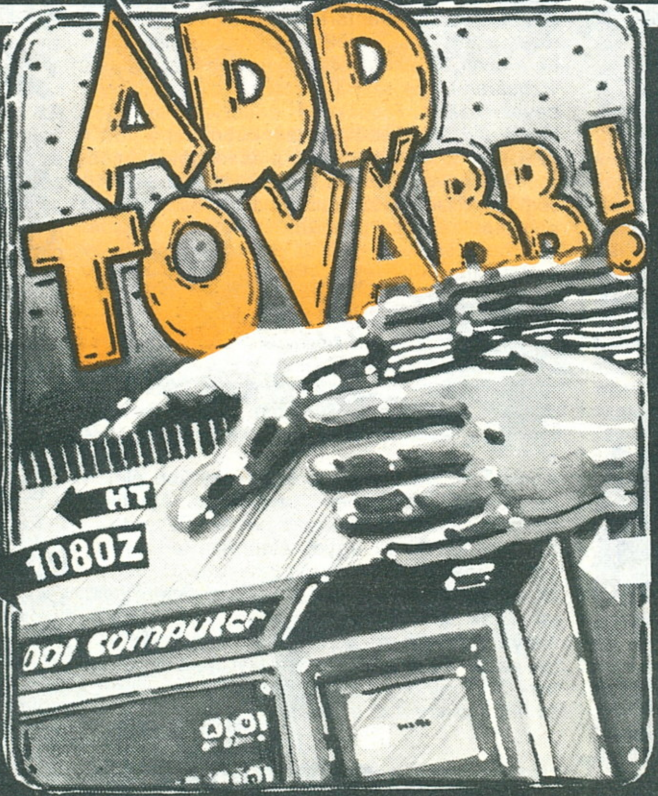
BIT-LET



Néhány hónappal ezelőtt egyik megyeszékhelyünkön egyik szervezője voltam egy olyan kulturális eseménynek, amelyhez szükségünk volt húszegynéhány HT típusú számítógépre. Gondoltam, ez igazán nem állíthatja nagy feladat elé a szervezőket, hiszen ez az 1-es számú iskolaszámítógép, s kifejezetten azért írattuk erre a gépre a szükséges programot, mert úgy véltük, még mindig kisebb gond ebből a típusból összeszedni annyit, mint mondjuk Commodore-ból, vagy Spectrumból. Nos a helyi szervezők némi küszködés után össze is szedték a gépeket. Meg kell mondanom, nagyon lehangoló volt már maga a gépek látványa. Ránézésre kiderült ugyanis, hogy legalább negyed részük rossz lesz. Letört gombok, hiányzó billentyűk, leszakadt csatlakozók.

A kipróbálás igazolta a látványt. A fizikailag is hiányos gépeken kívül is akadtak még el nem induló, semmilyen programot saját magnóval beolvasni nem tudó, félképernyővel bejelentkező gépek, amelyek a nem használhatók polcára kerültek. Hál'istennek a rendezők előrelátóak voltak, s kellő mennyiségű tartalék gépet is hozattak, így végül is némi javítgatás, barkácsolás után rendelkezésünkre állt a kellő mennyiségű működő gép.

Azután a rendezőkkel való beszélgetésben kiderült, hogy ezeknek a gépeknek egy része már hónapok óta használaton kívül van. Ott porosodik az iskola szertárában, vagy épp valamelyik szekrényben. No nem azért, mert a diákok nem tanulnak már számítástechnikát az adott iskolában, kivételesen nem is azért, mert elzárják előlük a gépeket. Nem. Épp ellenkezőleg. Az történt, hogy az iskolában jól dolgozó számítástechnikai szakkör újabb, modernebb gépeket kapott a tanácstól, az iskolától, a patronáló vállalattól. „Talán ugyanennyi Commodore-t könnyebben összeszedtünk volna!" – mondta az egyik szervező.

Meglepődtem egy kicsit, de végül is örültem. Azaz, hogy egyik szemem sír, a másik nevet az eset óta. Mindenképpen öröm, hogy az iskolák gyarapodnak. Öröm, hogy a diákok nyomására egyre több megye, egyre több iskola találja meg a módot, hogy minél több gépet vegyen, hogy minél színvonalasabb lehetőségeket biztosítson diákjainak. Azt is megértem, hogy ha a számítástechnikai körbe járók szívesebben ülnek a Commodore mellé, mint a HT mellé, ha tehetik. Még a pedagógust is megértem egy kicsit, aki egy idő után, látva, hogy a srácok inkább tétlenekednek, dumálnak, ha minden „jobb gép" foglalt, de az istennek nem ülnek már oda a HT mellé, tehát ha ezt látva a pedagógus feladja. Nem erőlteti a dolgot. Mondom, ezt is megértem, még ha el nem is fogadom, mert a HT azért szerintem jó tanulógép, elfelejteni nem érdemes, a kezdő szakkörnek meg különösen megfelel.

Ha mégis megértem a pedagógust, ez csak annak köszönhető, hogy tudom, „fárasztó" valakit meggyőzni, hogy üljön villamosra, ha autó is van. De! és épp ez a lényege mondandómnak.

Azt bajosan hiszem, hogy ma már a számítástechnika oktatás fizikai, műszaki feltételei országszerte olyan jók, hogy megengedhetnénk magunknak azt a luxust, hogy gépeket szertárakban hagyjunk porosodni, hogy apró hibával rendelkező gépeket ne javíttassunk meg, hanem nyugdíjba küldjünk. Meg kellene tehát oldani, hogy minden különösebb következmény nélkül, mindenféle rosszalló megjegyzés vagy a legközelebbi géposztásból való kimaradás veszélye nélkül lehetőséget adjunk az iskoláknak, hogy a már nyugdíjba küldött gépeket könyvjóváírással vagy jelképes összegért átadhassák olyan iskoláknak, amelyek úgy vélik, hogy nálunk még igenis jó szolgálatot tennének ezek a gépek. Ennyi az ötlet. Hogy mindezt ki vehetné kézbe, ki lehetne a közvetítő, a gépeket begyűjtő és újra elosztó? Nem tudom. Talán nem is kellene ehhez semmiféle központi segítség. Mindössze a szándék, az elhatározás lenne a fontos. Mert úgy hiszem, minden számításba jöhető iskola találna a környéken jelentkezőt. Talán még arra sincs szükség, hogy nagy dobra verje a gépet továbbadó iskola tettét. Csak az lenne a fontos, hogy a gépek dolgozzanak. Eredeti helyükön vagy másutt – mindegy.

Angyalosi László

BELÜLRŐL

ötlet HÍRADALOM

CANON KÉPLEMEZ

A japán fényképezőgép-gyártó cég egy különleges újdonsággal lepte meg a fotópiacot. Legújabb fényképezőgép-konstrukciójában a hagyományos filmanyag helyett egy mágneses képlemezt helyezett el. A lehetséges ötven felvétel elkészítése után a készülék tulajdonosa a képlemezt kiemelve és lemezleolvasó egységbe helyezve egy csatlakoztatott színes tv képernyőn megjelentetheti a felvételeket. Természetesen mód van arra is, hogy egy színes nyomtató segítségével papírképek készüljenek közvetlenül vagy egy telefonvonal közbeiktatásával egyaránt. Az új Canon rendszer egyetlen hibája ma még elképesztően magas, mintegy 32 ezer dolláros ára.

PAPRIKA!

A Szegedi Paprikafeldolgozó Vállalat szőregi telephelyén érdekes, kísérleti paprikaőrlő-keverő rendszert üzemeltetnek. Számítógépbe vitt szín- és ízadatok, illetve paprikareceptek alapján a kívánalmaknak megfelelően állíthatók elő, kitűnő minőségű, édesnemes, rózsa, csemege vagy csípős paprikakeverékek.

BRIDGE

Különleges és mindeddig egyedülálló bridzseseményre került sor a közelmúltban. Mintegy hetven ország százezer bridzselője ült asztalhoz, és lejátszotta ugyanazt a huszonnégy bridzspartit. A hatalmas vállalkozást a japán Epson számítástechnikai cég anyagi támogatásával bonyolították le. A helyi eredményeket telexen vagy telefonon továbbították Párizsba, ahol Omar Shariff filmszínész és bridzs világbajnok elemzésével elkészítették a számítógépes világranglistát. A magyar bridzselők két nagy parti rendezésével csatlakoztak a világversenyhez; az egyikre Budapesten, a másikra Keszthelyen került sor.

TISZTA VÍZ!

A tiszta víz olcsóbb előállításához használnak számítógépet a nyíregyházi szennyvíztisztító telepen. Az új technológiai eljárás lényege, hogy a tisztítómedencébe számítógépvezérléssel adagolják az oxigént és eresztik be a szennyvizet. A korábbi állandó levegőztetéssel szemben most a számítógép vezérlésével – a pillanatnyi oxigéntartalom folyamatos mérése mellett – csak szükség szerint kapcsolnak be a levegőztető berendezések, és a szennyvízbeáramlás is szabályozott. Az új módszer bevezetésével, csökkenő energiafelhasználással napi 24 ezerről 30 ezer köbméterre nőtt az üzem kapacitása. Ehhez más megoldásban kb. 27 milliós beruházásra lett volna szükség. Így viszont a számítógépes rendszer csupán ötmillió forintba került.

ÍZTURMIX

Angliában, Európa egyik legkorszerűbb élelmiszeripari üzemében százhatvannégyféle nyersanyagból számítógép vezérlésű automata keveri ki az egyes élelmiszertermékek ízét, zamatát. A számítógépes eljárás nagy előnye az, hogy segítségével tökéletesen betarthatók az előírt technológiai folyamatok, illetve biztosítható a recept szerinti gyártás. Így aztán az elkészült élelmiszer hosszú időn keresztül mindig azonos minőségben, ízben és illatban kerül el a vásárlókhoz.

LEVELEZŐGÉP!

Speciális üzleti levelező-fordító számítógépes programot dolgoztak ki angol szakemberek. Mivel viszonylag szűk szókincs és kifejezéstípus határok között kell mozogni, lehetővé vált, hogy a számítógépes fordítórendszer nem ún. nyers, hanem végleges fordításokat készítsen, összesen huszonkét nyelvre. A gép annyira tökéletes, hogy a „levelező" személynek egyáltalán nem szükséges ismernie a fordítás nyelvét. Csupán kiválasztja és összeállítja az angol típusmondatokat, és hozzáadja a szükséges adatokat (cím, dátum, aláírás stb.). A többi már a gép dolga.

BIOMEMÓRIÁK

Észt kutatók olyan biológiai memóriát hoztak létre, amelynek felhasználásával jelentősen növelhető a számítógépek tárkapacitása. A szerves anyag élő sejtjébe lézersugárral írhatók be az információk. Ennek alapja a fotolumineszcencia, melynek révén a fotoérzékeny biológiai elem, mint például a fehérjeklorofil információtárolóként viselkedik. Az észt szakemberek meg is alkották az első ezen elven működő berendezéseket. Az új memória tárolókapacitása köbcentiméterenként több milliárd bit, míg a tárolás időtartama néhány hónap lehet.

SZOVJET PROGRAM

Az 1985/86-os tanévtől kezdődően Szovjetunió-szerte megindul az általános iskolások és a középiskolások számítástechnikai képzése. A bevezetésre kerülő új tantárgy neve: „Az informatika és a számítástechnika alapja." Az új tantárgy bevezetésével például az ország mintegy 140 ezer általános iskolájába több százezer számítógép biztosítása válik szükségessé. A program végrehajtásának irányítására, a bevezetésre kerülő tantárgy anyagának kidolgozására Informatikai és Számítástechnikai Kutató Intézet néven új tudományos intézet kezdte meg működését Novoszibirszkben.

OMNIBOT 2000

A robottechnika ipari felhasználása mellett egyre inkább hallani újabb és újabb, a háztartásban is alkalmazható robotcsodákról. Ilyen például a japán Omnibot 2000, amit már az NSZK-ban 1400 márkáért meg lehet vásárolni. A kis önjáró szerkezet programozható. Képes lépcsőt mászni és az utcasarkon befordulni. Egyik fő feladata lehet a szőnyegek kiporszívózása, de ügyesen rakodik, és még drága porcelántárgyakat is rá lehet bízni. Mindemellett nagyszerűen bevált mint gyermekfelügyelő is.

FINN KAPCSOLAT!

A külkereskedelmi adminisztrációs tevékenység korszerűsítésére, a KGST-ben létrejött megállapodás alapján magyar és finn vállalatok közös számítógépes táradatátviteli kísérletet indítottak be. A kísérlet része egy hosszú távú programnak, melynek célja az áruforgalom technikai feltételeinek a javítása. Az együttműködés kapcsán elsőként a Datorg Külkereskedelmi Adatfeldolgozó RT-nél és két finn vállalatnál épül ki a kísérleti rendszer, amely a kereskedelmi, szállítmányozási adatokat továbbítja.

ALBACOMP

Az Albacomp Számítástechnikai Kisszövetkezet az általános iskolai számítógépprogram mielőbbi megvalósításának elősegítésére Székesfehérvár általános iskolái részére huszon-

egy Commodore–16-os személyi számítógépet és huszonegy 1531 típusú Dataset adatrögzítő magnetofont ajándékozott. A mindössze huszonöt tagból álló kisszövetkezet nem mindennapos akciójával a segítségnyújtáson túlmenően magára irányította a figyelmet, ami reklámnak sem utolsó.

GÁZMÉRŐ

A Kőolaj- és Földgáztermelő Vállalat szegedi üzemének laboratóriumában japán gyártmányú, számítógépes vezérlésű gázmérő készüléket vettek használatba. Az új, úgynevezett gázkromatográfiás analizátor segítségével a korábbiakhoz képest pontosabban és kétszer gyorsabban lehet megállapítani a földgáz és különböző származékai minőségét.

VÉRELLÁTÁS!

TPA–11/440-es számítógépből és nyolc képernyős terminálból álló rendszert helyeztek üzembe az Országos Haematológiai és Vértranszfúziós Intézetben. A rendszer nyilvántartja az intézet véradóinak személyi adatait, vérjellemzőit, s hogy mikor milyen célra adtak vért. A magas színvonalú technika alkalmazása kizárja a tévedés lehetőségét. A gép mindegyik vérről legalább hetvenféle adatot regisztrál. Összességében a rendszer mintegy százmillió karakternyi információt tárol. Az új technikai eszköz segítségével az intézet munkatársai percek alatt megkapják azok nevét, akik az adott esetben az életmentő vért adhatják.



ÚJ!

Az Atari gépek legújabbika a 16/32 bit technológiára épül. Mint más ST sorozatú gépek, az Atari 1040 STF is kiválóan alkalmas szövegfeldolgozásra, adatbáziskezelésre, üzleti programcsomagok futtatására stb.
Az Atari 1040 STF jól futtatható számos, népszerű operációs rendszer alatt, mint például a CP/M. Az új számítógép 1024 K RAM-ot, 1 Mbyte-os beépített, 3 és fél collos, kétoldalas lemezmeghajtót, két nyomógombos egeret és beépített tápegységet tartalmaz. Ajánlott programnyelvek a BASIC és a LOGO. A rendszerhez 12 collos monochrome képernyő tartozik.

NASA

Az amerikai űrhajózási hivatal a NASA nekilátott a világ legnagyobb szuperszámítóközpontja megépítésének. Az egyedülállóan nagyteljesítményű számítástechnikai létesítményben, a legújabb szuper számítógépek rendszerét kívánják üzemeltetni. Már kísérleti jelleggel működik a Cray–XMP5, másodpercenként 50 millió műveletet végző gép. Rövidesen bekapcsolódik az első Cray–2 szuperszámítógép 250 millió művelet/másodperc sebességgel. Ezt újabb Cray–2-esek követik majd. Öt-hat év múlva pedig, olyan gépeket helyeznek üzembe, amelyek már 4 milliárd műveletet végeznek másodpercenként. A NASA vezetői bíznak benne, hogy a hatalmas beruházás jelentős gazdasági eredményeket hoz. Ezekkel a számítógépekkel lehetővé válik az újonnan tervezett repülőgép vagy űrhajók első prototípusainak elhagyása, szélcsatorna vizsgálata. Ugyanis ezzel a számítógéprendszerrel könnyen szimulálhatók a szilárd test körüli aerodinamikai áramlások. Ugyancsak nagy hasznot hozhat a rendszer segítségével szinte ideálisra megtervezett repülőgépek légellenállás csökkenéséből adódó üzemanyagmegtakarítás is.

ÉPÍTÉS

Az angol építőipari ágazatban egyre terjed a nagy építkezések előzetes számítógépi megtervezése. Egyre több szoftvercég fejleszt ki építőipari tervezőprogramokat. Vannak szoftverek például a „kritikus út" analízisére, vonalas diagramok módszerére, és így tovább. Ilyen számítógépes terv alapján épül napjainkban a Lloyd biztosítótársaság új, londoni székháza. A számítógépes programokat a Claremont Control cég szállította. A programokat Commodore 8096 típusú számítógépen futtatják.

FÚZIÓ!

Módosul a nagy amerikai számítástechnikai cégek listája. A harmadik legnagyobb, a Borroughs' egyesül a hetedikkel, a Sperryvel. Arról van szó tulajdonképpen, hogy a Borroughs' 4,8 milliárd dollárért felvásárolja a Sperry 58 millió részvényét, és így a harmadik helyről az első számú IBM utáni, második helyre lép. Az eddigi második, a Digital Equipment a harmadik helyre szorul vissza. A fuzionáló két vállalat tavalyi forgalma 10,5 milliárd dollár volt, nyereségük pedig több mint 1 milliárd.

fagylaltos szindróma



INTERRUPT RUTIN ZX SPECTRUMRA

1. lista

```
   1 REM ███ fagylaltos ███
  10 DEF FN k(x)=x+1-(x+1>uzletm
eret)*uzletmeret
  20 GO SUB 500
  30 GO SUB 100
  40 STOP
 100 REM ███ foprogram
 110 INPUT "Nev , x , vege >>>";
 LINE u$
 120 IF u$="" THEN  GO TO 110
 130 IF u$="x" THEN  GO SUB 2000
: GO TO 160
 140 IF u$="vege" THEN  GO TO 17
0
 150 LET s$=u$: GO SUB 1000: GO
TO 160
 160 GO TO 110
 170 RETURN
 500 REM ███ kezdeti ertekek
 510 LET uzletmeret=10
 520 LET oszlop=10
 530 LET keszsor=4
 540 LET keszsor=4
 550 LET kezdosor=keszsor+3
 560 LET vegsor=kezdosor+uzletme
ret+1
 570 LET kiszolgalas=1
 580 LET sorbaallas=0
 590 LET utolsotev=kiszolgalas
 600 LET elso=1
 610 LET utolso=uzletmeret
 620 LET tele=0
 630 LET normal=1
 640 LET ures=2
 650 LET nevhossz=15
 660 DIM a$(uzletmeret,nevhossz)
 670 DIM s$(nevhossz)
 680 DIM k$(nevhossz)
 690 LET z$="                   "
 700 GO SUB 800
 710 RETURN
 800 REM ███ kezdeti kiirasok
 810 CLS
 820 PRINT AT 0,5; BRIGHT 1;"Fag
ylaltos szimulacio"
 830 PRINT AT keszsor,oszlop; IN
VERSE 1;" Fagyit kapott "
 840 PRINT AT kezdosor,oszlop; I
NVERSE 1;" Sor eleje "
 850 PRINT AT vegsor,oszlop; INV
ERSE 1;" Sor vege "
 860 RETURN
1000 REM ███ jott valaki
1010 GO SUB 3000
1020 IF allapot=tele THEN  GO SU
B 5000: GO TO 1050
1030 LET utolso=FN k(utolso)
```

Akik dolgoztak már IBM PC-n, bizonyára örömmel vették észre, hogy amíg a gép az egyik parancsukat hajtja végre, addig már gépelhetik is a következőt. Egy ehhez hasonló dolgot csináltam a Spectrumra, amit „Billentyű”-nek neveztem el. A program jelenleg a következőket tudja:

– BASIC programok futása közben megjegyzi a lenyomott billentyűket, azok nem vesznek el.

– A tárolt billentyűk kódjait vissza tudja adni a BASIC programnak.

Mire jó ez az egész? Ha a programot már eléggé ismerjük, és tudjuk, hogy a következő kérdésre mit fogunk válaszolni, ezt begépelhetjük, mielőtt még a kérdés megjelenne. Játékprogramokban is jól használható, hiszen nem fordulhat elő, hogy a játékos lenyomott valami billentyűt, és az figyelmen kívül marad, mert a program éppen akkor nem figyelte a billentyűzetet, hanem például számolt.

Természetesen nem akárhány billentyűt tud megjegyezni a program. Egyszerre 32 olyan billentyű lehet, amit már lenyomtak, de a program még nem használt fel. Ha éppen 32 ilyen billentyű van, akkor az újabbak figyelmen kívül maradnak.

A program ismertetése előtt két másik programot mutatok be. Ezek a jobb megértést szolgálják. Az első program egy fagylaltosnál álló sort szimulál, és jó példát ad az ún. „sorban” állás probléma egy lehetséges megoldására. A második program interrupt rutinok készítésének kérdésával foglalkozik. Egy jól és könnyen használható módszert ismertet, amelynek a felhasználásával készült a cikk fő témájául választott program is.

Lássuk tehát a két előkészítő programot:

A fagylaltos

Egy cukrászmester nyit egy kis fagylaltozót. Egy meleg nyári nap az emberek olyan gyorsan özönlenek az üzletbe, hogy nem tudják őket azonnal kiszolgáìni. Sorba kell tehát állniuk. A fagylaltos mindig a sor elején állót szolgálja ki, aki ezután kimegy az üzletből. Tudjuk még, hogy hányan férnek el egyszerre az üzletben. Ha pont ennyien állnak sorba, akkor már senki sem állhat be a sor végére.

Írjunk egy programot, ami a fagylaltos forgalmát szimulálja. A program feladata az, hogy mindig tartsa nyilván az üzletben álló sort, és tudja, hogy kit szolgáltak ki utoljára. Bemenő adatai legyenek a következők:

„vége” – a program leállítása
„x” – a sorban az első embert ki kell szolgálni
egy név – az adott nevű embert be kell állítani a sor végére.

A programnak figyelnie kell arra is, hogy
a) üres üzlet esetén nem lehet senkit sem kiszolgálni;
b) tele üzlet esetén nem állhat be senki a sorba.

A program megírásához legelőször is tisztázni kell, hogy milyen módon tároljuk a sort (BASIC nyelven). Mivel tudjuk, hogy egyszerre hány ember lehet az üzletben, ezért egy ekkora tömb meg fog felelni a célnak.

Vezessük be a következő változókat:
üzletméret – hány ember fér el az üzletben,
névhossz – az emberek nevének maximális hossza,

```
1040 LET a$(utolso)=s$
1050 LET utolsotev=sorbaallas
1060 GO SUB 7000
1070 RETURN
2000 REM ███ ki kell szolgalni v
alakit
2010 GO SUB 3000
2020 IF allapot=ures THEN  GO SU
B 6000: GO TO 2050
2030 LET k$=a$(elso)
2040 LET elso=FN k(elso)
2050 LET utolsotev=kiszolgalas
2060 GO SUB 8000
2070 RETURN
3000 REM ███ milyen az uzlet all
apota
3010 LET feltetel=(elso=FN k(uto
lso))
3020 IF NOT feltetel THEN  LET a
llapot=normal: GO TO 3050
3030 IF utolsotev=kiszolgalas TH
EN  LET allapot=ures: GO TO 3050
3040 IF utolsotev=sorbaallas THE
N  LET allapot=tele
3050 RETURN
5000 REM ███ tele az uzlet
5010 PRINT #1; FLASH 1;" Tele az
 uzlet ! "
5020 PAUSE 50
5030 RETURN
6000 REM ███ ures az uzlet
6010 PRINT #1; FLASH 1;" Ures az
 uzlet ! "
6020 PAUSE 50
6030 RETURN
7000 REM ███ a sor kiirasa
7010 LET index=elso: LET j=kezdo
sor+1
7020 GO SUB 3000
7030 IF allapot=ures THEN  GO TO
 7080
7040 PRINT AT j,oszlop;a$(index)
7050 LET index=FN k(index): LET
j=j+1
7060 IF index=FN k(utolso) THEN
 GO TO 7080
7070 GO TO 7040
7080 FOR i=j TO vegsor-1
7090 PRINT AT i,oszlop;z$
7100 NEXT i
7110 RETURN
8000 REM ███ kiszolgaltak valaki
t
8010 PRINT AT keszsor+1,oszlop;k
$
8020 GO SUB 7000
8030 RETURN
```

a$ (üzletméret, névhossz) – a fagylaltosnál álló sor tárolására,
s$ – annak neve, akit be kell állítani a sor végére,
k$ – annak a neve, akit utoljára kiszolgáltak,
állapot – az üzlet állapota (üres, normál, tele).
Nézzük meg ezek után, hogy lehet a sort az a$-ban tárolni. A sorban első ember neve legyen az a$ (1)-ben, a másodiké az a$ (2)-ben stb. E megoldás jó is lenne, csak egy probléma van: ha valakit kiszolgálnak, akkor az egész sor előbbre lép eggyel. Ez azt jelenti, hogy az a$ elemeit előbbre kellene mozgatni. Ez nagy üzlet esetében roppant munkaigényes lenne. Sokkal jobb megoldás, ha az a$-t nem változtatjuk, csupán egy változó segítségével nyilván tartjuk, hogy a sorban az első az a$-ban hanyadik. Vezessük még be a következő két változót:
első – a sorban az első ember az a$ hanyadik elemének felel meg,
utolsó – a sorban az utolsó ember az a$-ban hanyadik.
Ezzel azonban még nem vagyunk készen. Nézzük csak meg a következő példát:
üzletméret=5; első=3; utolsó=5

```
a$= [ ][ ][a][b][c]
           ↑     ↑
         első  utolsó
```

Látjuk tehát, hogy hárman vannak az üzletben. Ha most jön valaki, és beáll a sorba, azt nem tudjuk az utolsó elem után betenni, hiszen az a$ öt elemből áll. Mondjuk egyszerűen azt, hogy az a$ (6) feleljen meg az a$ (1)-nek. Ekkor, ha jön egy „d" nevű ember, a helyzet a következő lesz:
első=3; utolsó=1

```
a$= [d][ ][a][b][c]
     ↑     ↑
  utolsó  első
```

Elkészíthetünk egy függvényt, ami az a$ minden eleméhez megadja, hogy mi az utána következő elem:
DEF FN k(x)=x+1−(x+1> üzletméret)*üzletméret
Az előző példában FN k(5)=1, FN k(1)=2, FN k(2)=3 ...
Most már csak a három alapfeladat megoldása van hátra:
1. ha jön valaki, és nincs tele az üzlet, akkor beállítani a sorba;
2. ha nem üres az üzlet, kiszlogálni a sorban elöl állót;
3. megállapitani az üzlet állapotát (üres, normál, tele)
Lássuk ezek megoldását:
1. LET utolsó=FN k(utolsó)
   LET a$ (utolsó)=s$
2. LET k$=a$ (első)
   LET első=FN k(első)
3. A példából jól látszik, hogy ha még egy ember beáll a sorba, akkor megtelik az üzlet. Ebben az esetben az első=FN k(utolsó).
Már csak az a kérdés, hogy mikor üres az üzlet. Ez a következőképpen határozzuk meg: vegyünk egy egyetlen emberből álló sort. Ha őt kiszolgálják, üres lesz az üzlet, azaz:

```
egy ember a sorban         kiszolgálják,          üres üzlet
[ ][ ][a][ ][ ]        ─────────────────>      [ ][ ][ ][ ][ ]
       ↑  ↑                                          ↑  ↑
    első  utolsó                               utolsó  első
```

Vagyis azt kaptuk, hogy első=FN k(utolsó). De ugyanezt a feltételt kaptuk arra is, hogy tele van az üzlet. Ennyi információ birtokában tehát nem tudunk dönteni. Kell még valamit tudnunk. Például azt, hogy utoljára mi történt: kiszolgálás vagy sorban állás. Így már eldönthető az üzlet állapota.
Lesz tehát még egy változónk:
utolsótev – mi történt utoljára (kiszolgálás, sorba állás)
A „fagylaltos program" az 1. sz. lista.
A három alapfeladat megoldása az 1000–2000, 2000–3000, 3000–4000 sorokban található.
Még egy észrevétel: Ez az ún. „sorban állás" feladat sok más esetben is előfordul. Például:
– a telefonközpontban a telefonvonalra várók is sorban állnak;
– a bányában sorakoznak a csillék, hogy megtöltsék őket.

Interrupt rutinok

Most nézzük meg, hogyan dolgozik a Spectrum!
A CPU szépen sorban végrehajtja az éppen futás alatt álló program utasításait. Minden utasítás végrehajtása előtt megvizsgálja, hogy nincs-e valami sürgős elvégeznivalója. Ezt a CPU-nak egy adott lábán megjelenő jel mutatja. Ha van ilyen sürgős dolog, akkor a CPU félreteszi a futó programot, és meghív egy ún. interrupt programot. Ez az interrupt program a Spectrum normál működése közben a 38H címen kezdődő gépi kódú rutin. Ez olvassa le a billentyűzetet, és állítja a belső órát. A Spectrumban ez a rutin minden 1/50 másodpercben végrehajtódik. Vagyis másodpercenként 50-szer kap a CPU egy olyan jelet, ami sürgős elintéznivalóra utal. Ezt a jelet nevezzük interruptnak (megszakítás). A megszakítás rutin lefutása után a CPU előveszi a félbeszakított programot, és azt folytatja, egészen egy újabb interruptig.
A Z80 CPU három különböző megszakítási módot ismer (0, 1, 2). Ezek közül most nézzünk meg kettőt!
1. Megszakítási mód:
A Spectrum normális esetben ezt használja. Ebben a módban interrupt esetén a CPU a 38H címen kezdődő gépi kódú programra adja a vezérlést.
2. Megszakítási mód:
Interrupt esetén a CPU kiszámítja a meghívandó program címét, majd ráadja a vezérlést.
A címkiszámítás a következő módon történik:
A cím 16 bitjéből a felső 8-at az I regiszter adja. Az alsó 8 bitre pedig 255

2. lista

```
; interrupt program

ireg    equ   48      ;I regiszter
pcim    equ   60208   ;prog cime
screen  equ   16384   ;kepernyo cim

        org   pcim          ;sajat
        push  af            ;  rutin
        ld    a,255
        ld    (screen),a
        pop   af
        rst   38h           ;ROM rutin
        reti

; bekapcsolas

bekap   ld    a,ireg
        ld    i,a
        im    2
        ret

; kikapcsolas

kikap   ld    a,3fh
        im    1
        ld    i,a
        ret

        end
```

3. lista

```
   1 REM ██ Vonal ██
  10 REM ██ Be - RAND USR 60218
  20 REM ██ Ki - RAND USR 60225
  30 CLS
  40 CLEAR 60207
  45 LET vege=-1
  50 LET cim=60208
  60 READ a
  70 IF a=vege THEN  GO TO 110
  80 POKE cim,a
  90 LET cim=cim+1
 100 GO TO 60
 110 RANDOMIZE USR 60218
 120 STOP
1000 DATA 245,62,255,50
1005 DATA 0,64,241,255
1010 DATA 237,77,62,48
1015 DATA 237,71,237,94
1020 DATA 201,62,63,237
1025 DATA 86,237,71,201
1030 DATA vege
```

Amikor elraktározol valamit a gép memóriájában, raktározd el saját memóriádban, hogy hova tetted! (Leo Beiser első számítógép-axiómája)

```
;         Billentyu
;     Interrupt rutin

ireg    equ   48     ;I regiszter
pcim    equ   60208  ;prog cime
hossz   equ   32     ;puffer hosz
flags   equ   23611  ;rsz valtozo
flags2  equ   23658  ;rsz valtozo
lastk   equ   23560  ;rsz valtozo
kiv     equ   0      ;kivesz
ber     equ   1      ;berak
uresp   equ   0      ;ures puffer
telep   equ   1      ;tele puffer
normp   equ   2      ;normal puff

        org   pcim

;***** interrupt rutin belepesi
;***** pontja
belep
        push  bc
        push  de
        push  hl
        push  af
        ld    hl,flags
        ld    a,(hl)      ;flags el-
        ld    (fment),a   ;mentese
        res   5,(hl)      ;nincs bil
        rst   38h         ;ROM rutin
        bit   5,(hl)      ;van bill?
        jr    z,nincs
        ld    a,(lastk)   ;ha van,
        call  berak       ;betenni a
        jr    ki          ;pufferbe
nincs   ld    a,(fment)   ; nincs
        ld    (flags),a
ki      pop   af
        pop   hl
        pop   de
        pop   bc
        ret

;***** bekapcsolas
bekap
        call  init
        ld    a,ireg
        ld    i,a
        im    2
        ret

;***** kikapcsolas
kikap   ld    a,3fh
        im    1
        ld    i,a
        ret

;**** beallitja a puffert uresre
init
        ld    a,1
        ld    (elso),a
        ld    a,hossz
        ld    (utolso),a
        ld    a,kiv
        ld    (uttev),a
        ret

;***** kivesz egy betut a sor
;***** elejerol, es visszaadja
;***** BASIC-be.
; INPUT : -
; OUTPUT: BC - a betu kodja. Ha
;         nincs lenyomott bil-
;         lentyu, akkor 0
kivesz
        call  vizsg
        cp    uresp
        jr    z,kures
        ld    hl,puffer   ;nem ures
        ld    a,(elso)
        ld    c,a
        ld    b,0
        add   hl,bc
        call  leptet
        ld    (elso),a
        ld    a,(hl)      ;betu A-ba
        jr    kveg
kures   ld    a,0         ;nincs betu
kveg    ld    b,0         ;vissza
        ld    c,a         ; BASIC-be
        ld    a,kiv       ;utolso
        ld    (uttev),a   ;teendo
        ret   ;           ;tarolasa

;***** berakja a betut a puffer
;***** vegere
; INPUT : A - a betu kodja
; OUTPUT: -
berak
        ld    d,a         ;elmentes
        call  vizsg       ;tele a
        cp    telep       ;puffer?
        jr    z,btele
        ld    a,(utolso);nem tele
        call  leptet
        ld    (utolso),a
        ld    hl,puffer
        ld    c,a
        ld    b,0
        add   hl,bc
        ld    a,d         ;betu a
        ld    (hl),a      ;pufferbe
btele   ld    a,ber       ;utolso
        ld    (uttev),a ;teendo
        ret   ;           ;tarolasa

;***** megvizsgalja a puffer
;***** allapotat
; INPUT : -
; OUTPUT: A=telep, ha tele van,
;         A=uresp, ha ures,
;         A=normp, ha egyik sem
vizsg
        ld    a,(elso)
        ld    b,a
        ld    a,(utolso)
        call  leptet      ;elso =
        cp    b        ;FN k(utolso)?
        jr    nz,vnorm
        ld    a,(uttev)   ;utoljara
        cp    kiv         ;kivolg?
        jr    z,vures
vtele   ld    a,telep     ;tele puff
        jr    vki
vures   ld    a,uresp     ;ures puff
        jr    vki
vnorm   ld    a,normp     ;norm puff
vki     ret

;***** meghatarozza a puffer
;***** kovetkezo elemet. Az
;***** utolso utan az elso
;***** kovetkezik !!!
; INPUT : A - a jelenlegi elem
; OUTPUT: A - az uj elem
leptet
        inc   a           ;A=A+1
        cp    hossz+1     ;utolso u-
        jr    nz,lveg     ;tani?
        ld    a,1         ;akkor A=1
lveg    ret

elso    ds    1           ;bern elso
utolso  ds    1           ; az utolso
uttev   ds    1           ;utolsotev
fment   ds    1           ;flags ment
puffer  ds    hossz       ;puffer
        end
```

5. lista

```
   1 REM ■ BillenPyu ■
  10 REM ■ Be=RAND USR 60245
  20 REM ■ Ki=RAND USR 60255
  30 REM ■ Init=RAND USR 60262
  40 REM ■ Char=RAND USR 60278
  50 CLEAR 60207
  60 LET cim=60208
  70 LET vege=-1
  80 READ a
  90 IF a=vege THEN  GO TO 130
 100 POKE cim,a
 110 LET cim=cim+1
 120 GO TO 80
 130 PAPER 7: INK 0: CLS
 135 PRINT "Most fut egy ciklus
500-ig."
 140 PRINT "Kozben nyomkodd a bi
llentyuket!"
 150 RANDOMIZE USR 60245
 160 FOR i=1 TO 500
 170 PRINT AT 3,15;i
 180 NEXT i
 185 RANDOMIZE USR 60255
 190 PRINT "Vege a ciklusnak. Mo
st beolvasom"
 200 PRINT "a ciklus futasa kozb
en lenyomott"
 210 PRINT "billenyuket, de maxi
mum 32-t."
 220 PRINT
 225 PAPER 0: INK 7
 230 LET a=USR 60278
 240 IF a=0 THEN  GO TO 265
 250 PRINT CHR$ a;
 260 GO TO 230
 265 PAPER 7: INK 0
 270 PRINT
 280 PRINT "Ennyi volt."
 290 STOP
1000 DATA 197,213,229,245,33
1005 DATA 59,92,126,50,229
1010 DATA 235,203,174,255,203
1015 DATA 110,40,8,58,8
1020 DATA 92,205,155,235,24
1025 DATA 6,58,229,235,50
1030 DATA 59,92,241,225,209
1035 DATA 193,201,205,102,235
1040 DATA 62,48,237,71,237
1045 DATA 94,201,62,63,237
1050 DATA 86,237,71,201,62
1055 DATA 1,50,226,235,62
1060 DATA 32,50,227,235,62
1065 DATA 0,50,228,235,201
1070 DATA 205,187,235,254,0
1075 DATA 40,19,33,230,235
1080 DATA 58,226,235,79,6
1085 DATA 0,9,205,218,235
1090 DATA 50,226,235,126,24
1095 DATA 2,62,0,6,0
1100 DATA 79,62,0,50,228
1105 DATA 235,201,87,205,187
1110 DATA 235,254,1,40,18
1115 DATA 58,227,235,205,218
1120 DATA 235,50,227,235,33
1125 DATA 230,235,79,6,0
1130 DATA 9,122,119,62,1
1135 DATA 50,228,235,201,58
1140 DATA 226,235,71,58,227
1145 DATA 235,205,218,235,184
1150 DATA 32,15,58,228,235
1155 DATA 254,0,40,4,62
1160 DATA 1,24,6,62,0
1165 DATA 24,2,62,2,201
1170 DATA 60,254,33,32,2
1175 DATA 62,1,201,201,vege
```

BIT-LET

fagylaltos szindróma



INTERRUPT RUTIN ZX SPECTRUMRA

kerül (csak speciális bővítések nélkül!). Az így kapott cím még nem a meghívandó rutinnak a címe! A CPU megnézi, hogy az így kiszámított címen mi található. Az ott levő 2 byte adja meg végül a meghívandó rutin címét.

Megvan a lehetőségünk, hogy olyan gépi kódú programot írjunk, amit a CPU minden 1/50 másodpercben végrehajt. Ezt azonban csak a 2. megszakítási mód használatával tudjuk elérni. Két dolgot kell ehhez megcsinálnunk.

1. Meghatározni egy olyan címet, ahová az interrupt programot elhelyezhetjük, és kiszámítani, hogy ehhez az I regiszter milyen értéke tartozik.
2. Átállítani a CPU-t a 2. megszakítási módba.

Az első feladat megoldásához a következő táblázat nyújt segítséget. Megadja az egyes I regiszter értékekhez tartozó programcímeket. A táblázat a ROM tartalma alapján készült. Nem tartalmaz minden értéket, csak azokat, amelyek a RAM végére mutató címet határoznak meg:

I regiszter tartalma	interrupt program kezdőcíme
1	52818
9	65129
11	58888
12	53183
16	51984
45	55781
48	60208

A táblázat felhasználásával a következőképpen lehet interrupt programot készíteni:

a) Keresünk a táblázatban egy megfelelő címet. (Legyen ez a pcím.) Megírjuk az assembly programot, ami ORG pcím-mel kezdődik:

```
ORG pcím        ;ide kell helyezni a programot
(a program)
RETI            ;visszatérés a megszakításból.
```

b) Megírjuk az interrupt módot átállító (bekapcsoló) rutint. Ehhez megnézzük a táblázatban a pcím-hez tartozó I regiszter értéket. Legyen ez ireg

```
bekapcs ld a, ireg
        ld i, a; az I regiszter beállítása
        im 2; az interrupt mód átállítása.
```

c) Megírjuk a kikapcsoló rutint is. Ez visszaállítja az 1. megszakítási módot.

```
kikapcs, ld a, 3FH
        im 1
        ld i, a
        ret
```

d) Végül az egészet lefordíttatjuk az assemblerrel, és betöltjük a programot a memóriába (ne felejtsük el a CLEAR pcím-1-et!). Ekkor a
RANDOMIZE USR BEKAPCS és a
RANDOMIZE USR KIKAPCS segítségével ki/be tudjuk kapcsolni az interrupt programunkat.

Több program használ saját interrupt rutint. Ezek közül talán kettőt említenék:

ORA – Beépített óra. A képernyő jobb fölső sarkában állandóan látható a pontos idő.

RESET – Egy olyan BREAK, ami leállítja a gépi kódú végtelen ciklust is. Erre a CAPS+SPACE nem képes. A CAPS+ENTER billentyűket használja.

Lássunk egy nagyon egyszerű példát (2–3. lista). Annyi az egész program, hogy a képernyő bal fölső sarkába egy kis vonalat rajzol. Ezt azonban másodpercenként 50-szer, így a vonalat nem lehet eltüntetni (CLS, PRINT, ...). A program assembly nyelvű listáját, és BASIC változatát is közlöm. Az előbbi jobban érthető, az utóbbi pedig könnyen kipróbálható.

Még egy megjegyzés:

Van egy ún. DI utasítás, ami közli a CPU-val, hogy az interrupt kéréseket hagyja figyelmen kívül. Olyan gépi kódú programmal együtt, ami ezt használja, az interrupt rutinunk nem fog működni.

A DI párja az EI, ami újra megengedi a CPU-nak, hogy figyelembe vegye az interruptokat.

Billentyű

Miért volt szükség erre a két bevezető programra?

A fagylaltosnál sorban álló emberek ugyanúgy viselkednek, mint a „Billentyű" programban a beolvasásra váró karakterek:
– beállhat valaki a sor végére (lenyomnak egy billentyűt);
– eltávozhat valaki a sor elejéről (felhasználják a karaktert).

Az interrupt rutinok készítésének megismerése pedig azért volt hasznos, mert maga a „Billentyű" program is egy ilyen rutinra épül. Ezek után remélhetőleg senkinek sem okoz nehézséget a program megértése. Használatáról szólnék még egy keveset:

RANDOMIZE USR 60245 – a program bekapcsolása, az eddig elraktározott billentyűk törlése.
RANDOMIZE USR 60262 – az elraktározott billentyűk törlése.
RANDOMIZE USR 60255 – a program kikapcsolása.
LET a=USR 60278 – a következő billentyű lekérdezése. A lenyomott betű a CHR$ a volt. Ha a=0, akkor nem volt lenyomva egyetlen billentyű sem.

A program listája a 4. listában, a használatát bemutató BASIC program pedig az 5. listában „olvasható".

Gräff Zoltán, 3300 Eger, Egészségház u. 11.



Keserű Cukor Clive Lovagnak...

Még a mikroszámítógépek iránt ritkán érdeklődő magyar napisajtóba is bekerült a hír, hogy 1986. április 7-én, hétfőn Sir Clive Sinclair 5 millió angol fontért eladta ha nem is a lelkét, de majdnem: minden eddigi számítógépének szerzői jogait, gyártási és értékesítési jogát, és még a SINCLAIR márkanév használatát is. A vevő hazánkban kevésbé ismert: az Amstrad számítógépes és szórakoztató elektronikai cég (termékeit egyes nyugat-európai országokban Schneider néven forgalmazzák), Alan Sugar vezetésével. Mivel „sugar" angolul cukrot jelent, a brit sajtó szúrós tréfákat engedett meg magának Clive lovag keserű cukráról. Most, hogy a felvert por már kissé leülepedett, további információk is rendelkezésre állnak, és a lassúbb átfutású szaksajtóban is napvilágot láttak az első értékelő elemzések, érdemes magyar szemmel is áttekinteni a Sinclair-ügyet. Hogyan jutott odáig, hogy bagóért kelljen eladnia birodalmát riválisának (bagóért, hiszen cégét még két évvel ezelőtt 135 millió fontra értékelték!)? És főképpen mit várhatnak a jövőben a magyar ZX81, Spectrum és QL tulajdonosok, általában a mikrogépek jelenlegi vagy jövőbeni használói?

Mint a Sinclair gépek régi használója és a Spectrumig bezárólag barátja is, hadd idézzek négy évvel ezelőtti írásomból: „Utólag nyilvánvalónak tűnik, hogy a siker három titka a leleményes műszaki megoldásokban, a merész elhagyásokban és egyszerűsítésekben, valamint a tömeggyártásban rejlik. ...Míg a konkurencia a jelek szerint nem tudott kitörni az előző évek műszaki béklyóiból, Sinclair merész húzással mindent elhagyott gépéről, ami nem volt okvetlenül szükséges, felismerve, hogy a tömeggyártású számítógép csak tömegcikk áron kerülhet piacra. Sinclairék arra számítottak, hogy a vevő, akit megbabonáz a lehetőség, hogy igazi számítógéphez juthat 100 fontért – Amerikában 200 dollárért – megbocsát kisebb kényelmetlenségeket, hiányokat." (HVG, 1982. szept. 18.) Bár a cég csillaga még két-két és fél évig emelkedett (ami az anyagiakat illeti), „utólag nyilvánvalónak tűnik", hogy Sinclair akkor, a Spectrum átütő sikerű piacra dobásakor volt mikroszámítógépes pályája csúcspontján.

A Spectrum alapgépet még csak szállítási határidőproblémák nyomorították, meg egy-két ROM-beli poloska és hardver foltozgatás. Feledtették azonban ezeket a gép – sokszor és részletesen elemzett – értékei, és a néhány hónap alatt felfutó, máig irigyelt program- és hardverkiegészítő kínálat. A microdrive már éves késéssel került piacra, és máig nem tudta elfogadtatni magát, mint programtároló. (Ebben közrejátszott az is, hogy a játékprogramok jelentős részét csak bonyolult trükkökkel lehet microdrive-ra másolni, mivel memóriaterületük átfed.) Jellemző, hogy tudomásom szerint az első olyan komoly Spectrum program, mely kizárólag microdrive cartridge-en kerül forgalomba (a Tasword III), csak idén került piacra. Egyértelmű kudarcnak bizonyult az Interface II, mely botkormány csatlakozást kínált (de egyénieskedő módon az „ipari szabványtól", a Kempston-tól eltérő felállásban), és ROM kazettákat, amelyekből alig néhány készült azután. Amerikában Sinclair partnere, a Timex hozta ki a Spectrum egy változatát TS 2086 néven. Nagyot bukott vele, és ki is szállt az üzletből. (Vigaszként említhető, hogy a gépet Portugáliában, sőt újabban a hírek szerint Lengyelországban is tovább forgalmazzák.)

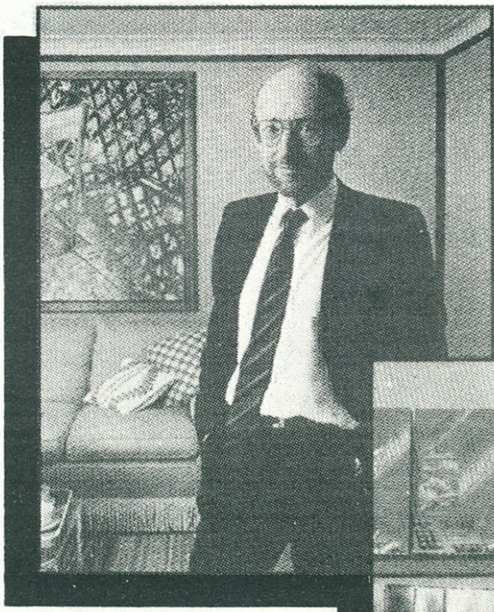
A QL, a következő Sinclair gép megosztotta a véleményeket, egyértelmű volt viszont, hogy az indulás szégyenletesen sikerült. Többszöri halasztás után végül egy olyan gép került piacra, melyből kilógott egy kis dobozka, mert úgymond nem fért el benne... A kiforrottabb változat megjelenése után ezeket a gépeket vissza is cserélték. Tény az is, hogy a QL-lel nem sikerült folytatni a ZX80-nal elkezdett a piaci sikersorozatot. Hiába a kiváló processzor (igaz, fapados változatában), hiába sok más tökéletes megoldás, és az értékes kapcsolt szoftver. A QL „olcsó gépnek" drága volt (400 fonttal indult, de ha valaki igazán élvezni akarta előnyeit, további 300 fontért kellett monitort is vennie hozzá!), „drága gépnek" viszont primitív, főleg a még mindig nem „igazi" billentyűzet és a QL-hez már végzetesen nem odaillő microdrive-ok miatt. A Spectrum+ és idén a Spectrum 128 piacra dobása már nem tudta a hanyatlást megállítani. A brit házi számítógéppiac még mindig jelentős, de ma már csak fele a korábbi csúcsnak. 1984 karácsonya után Sinclairnek már dömpingáron kellett szabadulnia felhalmozódott készleteitől. Az előző év 14 milliós profitja tavalyra 18 milliós veszteséggé változott. Lehetséges, hogy az otthoni (főleg játék) számítógép piaca telítődik? Angliában becslések szerint már minden negyedik családban van ilyen gép. Meddig lehet még újabb vevőket találni? Lehet, hogy a számítógép tömegcikkpiaca betöltötte feladatát, és átadja helyét a ténylegesen (hasznosan) használt, például szövegszerkesztő gépeknek? (Az Amstrad sikere erre utalhat.)

Nekünk, magyaroknak ideális gépet teremtett meg Clive lovag. A magyar turista két okból is olcsó alapgépet keresett: kevés volt a dollárja, fontja, márkája, schillingje, és félt a vámtól. Ne feledjük (hogy is felejthetnénk...), hogy a „vámérték" bűvészmutatványa még ma is a hivatalos árfolyamon átszámított beszerzési ár két-háromszorosára tornássza fel azt az alapot, amelynek akkor 40 (!), ma 15 vagy 30%-át kell fizetnünk. Sinclair „meztelen" gépet szállított, melyet vámmatematikusaink minden ügyessége is csak ideig-óráig tudott a bűvös 25 000 forint fölé értékelni. Tévét, magnót a meglevőből, hazai gyártmányt is csatlakoztathatunk hozzá. Szövegszerkesztéshez barkácsoltunk billentyűzetet. Memóriabővítéshez csipeket vettünk, és házilag dugdostuk a foglalatba. Interfészeket, botkormányt, microdrive-ot, netán „igazi" lemezmeghajtót hónapok, évek múltán külön vettünk, mire megint össze tudtuk spórolni a pénzt. És közben programoztunk és játszottunk. És akadtak olyanok is (senki ne vegye magára!), akik nem átallották védett programokat TOLVAJ MÓDJÁRA lemásolni! Ejnye! Mindent egybevetve, Sinclair meghódította a lapos zsebű magyarokat. Csak akkor fordult meg a szél, amikor a szerény magánvásárlók mellett feltűnedeztek a milliókat nem, de (forint) tízezreket könnyű kézzel kiadó közületi használók. (De ez már a hazai Commodore story, azt zengje más dalnok.)

Miért nem vált, Nyugat-Európához hasonlóan, hazánkban is sikerré az Amstrad-Schneider? Talán azért, mert szolid, de fantáziátlan gép? A C64-nél (magánvélemény!) nehezen lehet fantáziátlanabb, körülményesebb gépet elképzelni, az mégis sztár lett Hunniában. Az Amstrad éppen azért nem lett magyar siker, amiért a vad Nyugaton keresik: ugyanis minden bele van építve. A komplett gépbe monitor, magnó vagy diszk is tartozik, újabban a Joyce-ot printerrel kapcsolva árulják. A nyugati középrétegbeli, számító-

géphez nem értő vásárlónak mindez ideális. Amsrad Koreában gyártat; a kombináció ára lényegesen olcsóbb, mintha ezeket külön vásárolnák; és nem is kell drótokat dugdosni, vacakolni. A magyar vásárló viszont nehezen tud egyszerre kiizzadni ennyi valutát, és ha mégis, akkor itthon MNB-engedélyért szaladgálhat, és nyögheti a minimálisan is tízezer forint körüli vámot. De nem jár jobban a közületi vevő sem, hiszen a legolcsóbb Amstrad ára is túllépi a közületriasztó 50 ezret, és még nem mindenütt tudják, hogyan kell ilyenkor számlázni.
Magyar szemmel nézve tehát amikor Amstrad megvette Sinclairt, két gyökeresen különböző számítógépes filozófia került egy fedél alá. A nagy kérdés ezután G. B. Shaw híres anekdotájának módjára így fogalmazható meg: vajon a jövendő Amstrad–Sinclairek olyan ötletesek lesznek, mint az Amstrad, és olyan precízek és megbízhatóak, mint a Sinclair, vagy sikerül sinclairi szellemességű és amstradi megbízhatóságú számítógépet produkálni, sinclairi áron!? (Mint ismeretes, a bölcs G. B. Shaw azzal hűtötte le a tőle gyereket kívánó szépasszonyt, hogy hátha a gyermek olyan szép lesz, mint az apja, és olyan okos, mint az anyja...)
Mi az, amit már most is tudni lehet a jövőbeli gépekről? Sinclair nem lenne Sinclair, ha nem lennének most is vadnál vadabb pletykák bőviben. Tény, hogy Sugar azt nyilatkozta: nem kívánja tovább gyártani a QL-t, viszont, mint hangoztatta, hét évig kötelességük biztosítani a pótalkatrész-ellátást. Több cég is foglalkozik azzal, hogy átveszi és folytatja a QL gyártását. Van, aki igazi lemezmeghajtóval akarja gyártani, van, aki 68000-es, sőt 68020-as processzorral. Talán egy billentyűzettel is megszánnák. A baj csak az, hogy egy ilyen super-QL már olyan drága lehet, hogy a vevő akkor már inkább Atarit vesz, amely közben erősen felfutott, viszonylag nagyon olcsó, és bár később indult, máris sokkal jobb a szoftver ellátása, mint a QL-é. Tény az is, hogy a Spectrumból új változat készül, valószínűleg a 128-ast fejlesztik tovább. Találgatások vannak a változtatások mértékéről: lesz-e benne beépített magnó, lesz-e botkormány csatlakozója, lesz-e „rendes" billentyűzete. Sugar a microdrive-ot sem kívánja megtartani, és csak remélhetjük, hogy cartridge-ok továbbra is készülnek majd a meglevőkhöz. Ma már úgy hírlik, hogy a külföldi gyártási jogokat nem vette meg az Amstrad, így Portugáliában, Spanyolországban és (ismét: talán) Lengyelországban továbbra is készülhetnek a régi Sinclairek különböző változatai. Szinte biztos, hogy a Spectrum amstradosított változata az idei karácsonyi vásáron már kapható lesz; egyesek szerint már a szeptemberi PCW vásáron kihozzák. Biztos, hogy teljes marad a szoftver kompatibilitás: őrültek lennének lemondani a hatalmas programkincsről, ami Spectrumhoz készült.
Az igazi, szaftos pletykák azonban Sinclair saját tervezésű, de még nem kibocsátott gépeiről szólnak. Ott van a sokat emlegetett Pandora, egy hordozható gép(terv), melyet Clive bácsi megáldott volna microdrive-val és saját tervezésű lapos tévéképcsövével is. Most – úgy tűnik – sorra foszlanak le róla ezek a kétes áldások. Nem tudni viszont, mi marad akkor rendkívüli, hiszen számos Z80 alapú hordozható gépet ismerünk már. Sinclairtől persze meglepetés mindig jöhet. A Sinclair User júniusi száma egy ilyen meglepetés csodagéppel borzolja a kedélyeket, hol super-Spectrumnak, hol Loki-nak nevezve. (Loki az északi mitológiában a játékos isten.) Ha a leírásnak hinni lehet, a fantáziaképen ábrázolt csodagép mindent tud, ami szem-szájnak ingere. Z80H processzora az eddiginél kétszer gyorsabb, 7 MHz-es órajellel dolgozik. De hogy még gyorsabb legyen a képernyőkezelés, erre a célra, Amiga-módra egy külön tervezett csipet állítanak be, Rasterop néven. Ugyancsak saját hangcsipet kap, mely úgy szól, akár – eltalálták– egy Amiga, vagy egy szintetizátor. A Centronics printer kivételével szinte minden elképzelhető kimenetet kap a csodagép, azt is talán csak azért nem, hogy a perifériaépítők se haljanak éhen. Profi billentyűzete lesz, és fényceruzát adnak hozzá, és természetesen kompatibilis lesz (egy külön üzemmódban) a régi Spectrumprogramokkal. Mindezt a gyönyörűséget (hely hiányában csak a felét tudtam felsorolni, a többit tessék hozzáképzelni) mindössze 200 fontért kívánják utánunk vágni... Majd. Mert a csodagépnek csak egy baja van: hogy még a lap szerint is legkorábban egy év múlva készülhet el. Ennek ellenére a júliusi számban nagy körkérdést intéztek a vezető brit játékprogramozókhoz: szeretnének-e egy ilyen gépre programot írni? Persze, szeretnének. Bár a hagyományos brit hidegvér nem hagyja cserben őket, és válaszaik tele vannak „ha" kezdetű mellékmondatokkal. Egyikük pedig ki is mondja: a ma 1900 fontba kerülő Amiga teljesítményét 200 fontért meghaladni – túl szép ahhoz, hogy igaz legyen. Sinclair ugyan sokszor és sokféleképpen meglepett már bennünket – egyelőre azonban azt sem lehet tudni, hogy a Lokit, ha lesz, ki fogja befejezni – Sinclair vagy Sugarék emberei.
A Lokiról terjengő hírek számomra kicsit olyanoknak tűnnek, mint a 48-as szabadságharc után a Petőfi életbenmaradásáról tudósító híradások. A mikroszámítógépesek tábora nem tudja elhinni, hogy vége van egy korszaknak, hogy nem kapunk több furfangos gépcsodát Clive bácsitól. (Sinclair a hírek szerint több megabyteos óriásszelet-csipjén, valamint közelebbről meg nem nevezett „kommunikációs szervezeteken" kíván dolgozni.) Akárhogy is folytatódik a történet, mint a szurkolók, feledve az öngólokat és kihagyott helyzeteket, a magyar Sinclair-barátok kórusa is felhangozhat: Szép volt, fiú! Szép volt, Sir Clive!

Szekfű András



SIR CLIVE SINCLAIR



ALAN SUGAR



ötlet

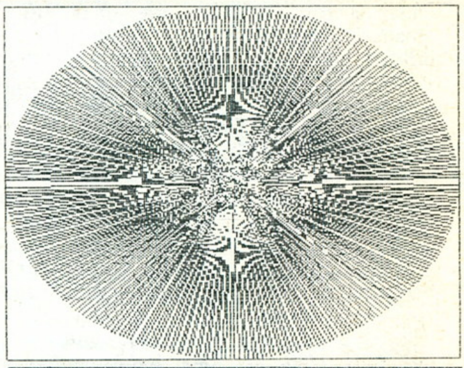


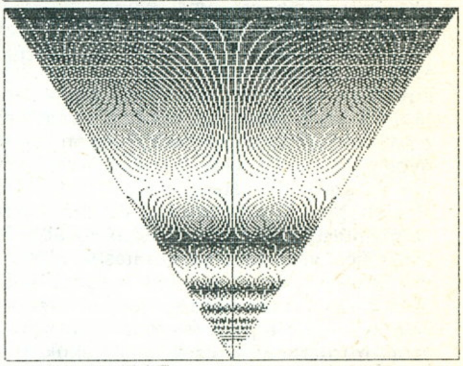
Mostanában egyre több, a TV Computerrel kapcsolatos kérdést kapunk. Némi fejtörés után – vajon ki válaszolhatna ezekre? Úgy döntöttünk, hogy megkeressük a legilletékesebbeket. Így hát a gép fejlesztésében részt vett fiatal mérnökökkel „kötöttünk egyezséget", miszerint mostantól kezdve ellátják olvasóinkat információkkal.

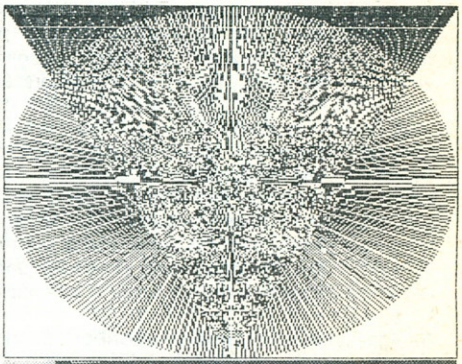
ELSŐ KÉZBŐL

A TV COMPUTERRŐL

```
 10 POKE 2842,127:RUN 20     ! Csak 64kB-os TVC-hez kell a 10.sor !
 20 !-------------------------------!
 30 !                               !
 40 !  TV-COMPUTER   1.sz. program  !
 50 !                               !
 60 !  Képernyötartalom elmentése   !
 70 !     és visszaállitása.        !
 80 !                               !
 90 !-------------------------------!
100 !
110 DATA 243        !   DI
120 DATA 62,80      !   LD    A,50h    ;page=UUVS
130 DATA 50,3,0     !   LD  (3),A
140 DATA 211,2      !   OUT (2),A
150 DATA 251        !   EI
160 DATA 201        !   RET
170 !
180 DATA 33,0,128   !   LD   HL,8000h
190 DATA 17,0,26    !   LD   DE,1A00h
200 DATA 1,0,60     !   LD   BC,3C00h
210 DATA 237,176    !   LDIR
220 DATA 201        !   RET
230 !
240 DATA 33,0,26    !   LD   HL,1A00h
250 DATA 17,0,128   !   LD   DE,8000h
260 DATA 1,0,60     !   LD   BC,3C00h
270 DATA 237,176    !   LDIR
280 DATA 201        !   RET
300 !
310 LOMEM 86*256                ! <--- kép elmentéséhez hely !
320 GRAPHICS 4:SET MODE 3:T=0
330 FOR I=1 TO 10:READ X:VID_ON$(I)=CHR$(X):NEXT I
340 VID=2+VARPTR(VID_ON$)
350 X=USR(VID)                  ! <--- VIDEO RAM bekapcsolása !
360 FOR I=1 TO 12:READ X:K_SAVE$(I)=CHR$(X):NEXT I
370 K_S=2+VARPTR(K_SAVE$)       ! <--- képelmentés kezdöcime !
380 FOR I=1 TO 12:READ X:K_LOAD$(I)=CHR$(X):NEXT I
390 K_L=2+VARPTR(K_LOAD$)       ! <--- képvisszaállitás kezdöcime !
400 !
410 SET INK 3;MODE 3
420 FOR I=0 TO PI STEP 0.02
430 RX=511*COS(I):RY=479*SIN(I)
440 PLOT 511-RX,479-RY;511+RX,479+RY
450 NEXT I
460 X=USR(K_S)                  ! <--- kép elmentése !
500 !
510 T=NOT T:SET PAPER 3+T;INK 1:CLS:PRINT AT 5,1:;
520 LIST 500-:SET INK 2+T
530 FOR I=1 TO 2000:NEXT I      ! <--- idözités !
540 X=USR(K_L)                  ! <--- kép visszaállitása !
550 FOR I=0 TO 1020 STEP 4
560 PLOT 512,956+T*956;I,-T*956
570 NEXT I
580 FOR I=1 TO 1000:NEXT I      ! <--- idözités !
590 IF INKEY$<>CHR$(13) THEN 510
```


1. ábra


2. ábra


3. ábra



1. táblázat: A "paging" bájt bitjeinek jelentése.

"paging" bájt 4. bit	3. bit	0. lapon címezhető memória (PAGE 0)	rövid jelölés
0	0	rendszer ROM fő része (SYSTEM ROM)	S
0	1	a TVC oldalán bedugható ROM-modul (CARTRIDGE ROM)	C
1	0	felhasználói RAM első 16 kB (USER RAM)	U
1	1	felhasználói RAM negyedik 16 kB (USER RAM +)	U+
"paging" bájt		**1. lapon címezhető memória (PAGE 1)**	**rövid jelölés**
-		felhasználói RAM második 16 kB (USER RAM)	U
"paging" bájt 5. bit		**2. lapon címezhető memória (PAGE 2)**	**rövid jelölés**
0		képernyő RAM (VIDEO RAM)	V
1		felhasználói RAM harmadik 16 kB (USER RAM)	U
"paging" bájt 7. bit	**6. bit**	**3. lapon címezhető memória (PAGE 3)**	**rövid jelölés**
0	0	a TVC oldalán bedugható ROM-modul (CARTRIDGE ROM)	C
0	1	rendszer ROM fő része (SYSTEM ROM)	S
1	0	felhasználói RAM negyedik 16 kB (USER RAM +)	U+
1	1	rendszer ROM bővítés része (felső 8 kB, ebből 4kB üres) (EXTENSION SYSTEM ROM) és bővítőkártyán levő RAM/ROM (alsó 8 kB) (EXPANSION CARD RAM/ROM)*	E

* Bővítőkártya kiválasztása a 3.port-ra kiírt bájt 6. és 7. bitjével történik.

2. táblázat: A memória-kiosztásokhoz tartozó "paging" bájtok.

"paging" bájt decimális	hexa		P0	P1	P2	P3
0	00		S	U	V	C
64	40		S	U	V	S
128	80		S	U	V	U+
192	C0		S	U	V	E
32	20		S	U	U	C
96	60		S	U	U	S
160	A0		S	U	U	U+
224	E0		S	U	U	E
8	08		C	U	V	C
72	48		C	U	V	S
136	88		C	U	V	U+
200	C8		C	U	V	E
40	28		C	U	U	C
104	68		C	U	U	S
168	A8		C	U	U	U+
232	E8		C	U	U	E
16	10		U	U	V	C
80	50		U	U	V	S
144	90		U	U	V	U+
208	D0		U	U	V	E
48	30		U	U	U	C
112	70	*****	U	U	U	S
176	B0		U	U	U	U+
240	F0		U	U	U	E
24	18		U+	U	V	C
88	58		U+	U	V	S
152	98		U+	U	V	U+
216	D8		U+	U	V	E
56	38		U+	U	U	C
120	78		U+	U	U	S
184	B8		U+	U	U	U+
248	F8		U+	U	U	E

***** Aktuális memória-kiosztás BASIC alatt

Rövidítések:
S - rendszer ROM fő része
V - képernyő RAM
U - felhasználói RAM
U+ - felhasználói RAM a 3. lapon
C - oldalt bedugható ROM-modul
E - rendszer ROM bővítés és bővítőkártya RAM/ROM
P0, P1, P2, P3 - 16 kB-os memórialapok

Aki már próbálta, tapasztalhatta, hogy a TV-Computer emberközeli számítógép. A Kezelési Útmutató alapján hamar megtanulható az egész képernyős programszerkesztés (full screen editor), a gép BASIC nyelvének tömör ismertetése pedig a BASIC Programozási Segédletben található. Mindez azonban kevésnek bizonyul, ha gépi kódban szeretnénk programozni. Cikksorozatunkban ismertetni fogjuk az alapvető ismereteket a TV-Computer gépi kódú programozásához.
A TV-Computer (továbbiakban: TVC) beépített programja két részre bontható, ebből az egyik rész a BASIC interpreter. A másik rész az operációs rendszer, amely az egyes logikai eszközöket kezelő, a konkrét input-output műveleteket végrehajtó rutincsomagból és az ehhez tartozó hozzárendelési struktúrából áll. Ezt a második részt a gépi kódú programok is használhatják. Következő cikkekben ismertetjük az operációs rendszer rutinjainak hívási módját, a rutinok rövid leírását, a hozzájuk tartozó rendszerváltozókat, az input-output perifériacímek (portok) kiosztását és funkcióit, a TVC memóriakezelését, valamint a gépi kódú programok BASIC-ba ágyazását.
Itt jegyezzük meg, hogy a közeljövőben jelennek meg a TVC hardvert és az operációs rendszert részletesen ismertető könyvek.

A TVC memóriái

A TVC-ben Z80 típusú 8 bites mikroprocesszor van, ami közvetlenül csak 64 kB memóriát tud címezni. A TVC már alapkiépítésben is ennél több memóriát tartalmaz (32 kB felhasználói RAM, 16 kB video RAM, 20 kB rendszer ROM), emellett lehetőség van további memóriák használatára is. A több memória elérése a memórialapozás (paging) segítségével történik. A hardver a processzor címtartományát négy, egyenként 16 kB-os részre osztja, ezek lesznek a memórialapok:

0. lap (PAGE0): 0–16383 azaz 0h–3FFFh.
1. lap (PAGE1): 16384–32767 azaz 4000h–7FFFh.
2. lap (PAGE2): 32768–49151 azaz 8000h–0BFFFh.
3. lap (PAGE3): 49152–65535 azaz 0C000h–0FFFFh.

Egy-egy lapon többféle memóriát is ki lehet választani. A kiválasztást egy hardver regiszter („page" regiszter) végzi, a megfelelő értéket a regiszterbe a 2. perifériacímen (port2) keresztül kell beírni. A 0. lapra négy választási lehetőségünk van (két bit), az 1. lapra egy (nincs bit), a 2. lapra kettő (egy bit), a 3. lapra pedig négy (két bit). A kiküldött byte („paging" byte) ezek alapján öt értékes bitet tartalmaz, a 0., 1. és 2. bit értéke közömbös.
Az egyes bitekkel kiválasztható memóriák az 1. táblázatban találhatók. Bővítőkártyán levő memória eléréséhez még külön ki kell választani a bővítőkártyát tartalmazó csatlakozót egy hardverregiszterrel. Ebbe a regiszterbe a 3. perifériacímre (port 3) kiküldött byte 6. és 7. bitje kerül. A kiküldendő bitekhez hozzá kell maszkolni a PORT03 rendszerváltozó 0–5. bitjét (címe: 2833=0B11h), és a rendszerváltozóba az új értéket vissza kell írni.
Alaphelyzetben, a BASIC futása alatt a memória négy lapjára az alábbiak vannak kiválasztva:

PAGE0:	PAGE1:	PAGE2:	PAGE3:
USER RAM	**USER RAM**	**USER RAM**	**SYSTEM ROM**

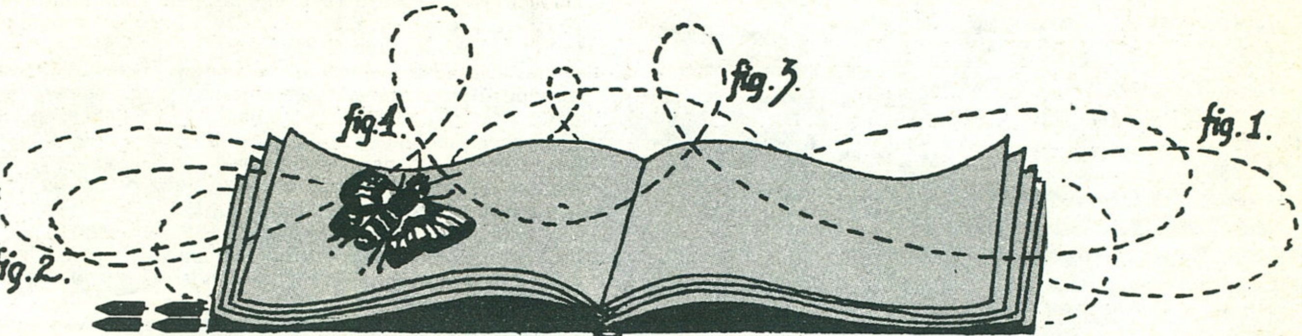
Röviden: UUUS.

Ezt a kiosztást a felhasználó által írt programok módosíthatják, sőt bizonyos esetekben módosítaniuk kell. Ilyen pl. az az eset, amikor a P3 alatti USER RAM területen USR rutinokat akarunk futtatni (ezt a RAM-területet a BASIC soha nem használja!), vagy a P2 területen levő VIDEO RAM tartalmát akarjuk kiolvasni vagy módosítani.
A memória lapozása a következő utasításokkal történik:

243	**F3**	**DI**
62,X	**3E X**	**LD A,X**
50,3,0	**32 03 00**	**LD (3),A**
211,2	**D3 02**	**OUT (2),A**
251	**FB**	**EI**

ahol X a „page" regiszter új értéke. Az új memóriaállapot beállítása az OUT utasítás hatására fog megtörténni. Vigyázni kell arra, nehogy a program „kilapozza maga alól a memóriát", vagy ha igen, akkor ez kiszámítottan történjen.
X lehetséges értéke megállapítható az 1. táblázat alapján is, de az egyszerűség kedvéért a 2. táblázatban közöljük a lehetséges értékeket a megfelelő memóriakiválasztás rövidített jelölésével. A közömbös biteket 0 értékűnek tekintettük.
Az LD (3),A utasítás a memória 3 című rekeszébe, azaz a P SAVE rendszerváltozóba elteszi a „page" regiszter aktuális értékét. Ennek célja, hogy az operációs rendszer rutinjainak meghívása, vagy megszakítás kiszolgálása után a hívás, illetve megszakítás előtti érték visszaállítható legyen.
Az 1. sz. program rövid példát mutat a lapozásra. A 100–300 sorok tartalmazzák a három gépi kódú rutint (az adatokat egyetlen DATA után is felsorolhatjuk). A 350-es sorban a 2. lapra bekapcsoljuk a VIDEO RAM-ot. A 32 kB-os alapgépben a 2. lapon nincs USER RAM, ezért többet nem kell lapozni. A 410–460-es sorokban készítünk egy hátteret a későbbi rajzok számára (1. ábra), és elmentjük a BASIC program elé. Ehhez a helyet a 310-es sorban biztosítottuk. Kilistázunk egy programkészletet (510–530), ezután elővesszük az elmentett hátteret a rajzhoz (540-es sor), és rárajzoljuk a 2. ábrán látható rajzot. Az eredmény a 3. ábrán látható (550–580-as sor). Ismétlés az 510-es sortól, ha nem nyomtunk RETURN-t (590-es sor).
A gyakorlatban ritkán kell az egész képernyőt elmenteni és visszaállítani, elegendő a képernyő egy részletével ezt megtenni. Ne feledjük, hogy az egész képernyő számára 15 kB helyet kell biztosítani elmentés előtt! 64 kB-os TVC-ben érdemesebb a képet a 3. lapon levő RAM-ba menteni, mert azt a BASIC interpreter soha nem használja.

Cseh Tibor



KÖNYVMOLY

Horváth–Révbíró: **Hetedhét ZX Spectrum** – Novotrade, 312 o., 95 Ft. Kazetta mintaprogramokkal: 234 Ft.
(A Hetedhét sorozat új kötete a korábbiaknál bővebb terjedelemben és eltérő formában – egy program fokozatos fejlesztésének bemutatásával – vezeti be az olvasót a gép kezelésébe.)

Operációs rendszerek időosztásos üzemmódjai I–II. Szerk.: Gerl Zsolt. – SZÁMALK, 510 o., 213 Ft.
(Az ESZR bázisú számítóközpontok felhasználói számára készült kötet azoknak nyújt segítséget, akiknek az elterjedt kötegelt feldolgozás mellett lehetőségük van az időosztásos interaktív üzemmód használatára is. Összegyűjtve ismerteti az IBM, illetve ESZR rendszerek körében, különböző operációs rendszerek alatt használható időosztásos alrendszereket, és kitér olyan kérdésekre is, melyek az egyes számítóközpontok rendszerprogramozói, karbantartói feladatkörébe tartoznak.)

Mikroszámítógép alkalmazási esettanulmányok. Szerk.: Dr. Rózsa Lajos. – SZÁMALK, 317 o., 156 Ft.
(Az MTA SZTAKI által kifejlesztett Mikroszámítógépes Folyamatirányító Berendezés alkalmazási lehetőségeit mutatják be a szerzők, akik maguk is részt vettek e technika kidolgozásában, illetve felhasználásában.)

Dr. Szentes János: **A szoftverminőség mérése** – SZÁMALK, 247 o., 107 Ft.
(A kötet a szoftverminőséget meghatározó tényezők modelljeit, mérőszámait, az ezek megállapítását szolgáló módszereket ismerteti, igazolva, hogy lehetséges a szoftverek minőségének objektív mérése, számszerűsítése.)

Bodor–Gerő: **A Commodore 64 programozásának gyakorlata 1. Alapismeretek** – SZÁMALK, 161 o., 55 Ft.
(A SZÁMALK C 64 sorozatának első kötete a kezdő felhasználóknak mutatja be a gép kezelését, programozásának alapjait.)

Ismerd meg a BASIC nyelvjárásait! Commodore 64, Commodore VIC 20, SHARP PC–1500. Szerk.: Kőhegyi János. – Műszaki Könyvkiadó, 187 o., 65 Ft.
(A „nyelvjárássorozat" – amely Donald Alcock: Ismerd meg a BASIC nyelvet! c. munkája alapján készült – legújabb kötete. A korábbiakhoz hasonló formában, három géptípus programozásának jellegzetességeit, eltéréseit tartalmazza.)

Fekete István: **Matematika és számítástechnika I.** – Műszaki Könyvkiadó, 205 o., 66 Ft (programfüzettel)
(A főiskolai tankönyvből gyakorlati jelentőségű matematikai problémák számítógéppel segített megoldását sajátíthatják el az üzemmérnök hallgatók.)

Commodore Plus/4 Felhasználói kézikönyv – Novotrade, 160 o., 99 Ft
(A gép használatának referenciaértékű ismertetése az eredeti Commodore-kiadvány fordításában.)

Commodore Plus/4 A beépített programok kezelése – Novotrade, 154. o., 99 Ft.
(Az előző kötethez kapcsolódva, a Szövegszerkesztő, Számviteli tömb, Grafika és Adatbázis programok használatának bemutatása.)

Dr. Szentes János: A szoftverminőség és mérése – Számalk, 247 o., 107 Ft.

Az informatika új ága van születőben. A szoftverek minőségének mérése, ha valamilyen tudományos kategóriába akarjuk sorolni, „inter-interdiszciplínának" tekinthető, hiszen maga az informatika is a tudományok köztes területén helyezhető el. Ez az ága pedig a számítástechnika, a méréselmélet és a gráfelmélet korábbi eredményeit használja fel, hogy a szoftvertermékek minőségére jellemző mutatókat határozzon meg.

A laikus olvasó, akit általánosságban érdekel a számítástechnika (illetve az informatika) – belelapozva Szentes János könyvébe – visszariadhat a kötet lapjain minduntalan szembeötlő matematikai kifejezésektől, formuláktól. Ha viszont rendelkezik némi gráfelméleti és méréstudományi jártassággal – akár csak alapfokon is –, és rászánja magát a kötet alapos áttanulmányozására, akkor érdekes és hasznos ismereteket szerezhet az informatikának e kialakulóban levő részterületéről.

Miért is fontos annak meghatározása, hogy az egyes szoftverek minőségét meghatározó jellemzők milyen értékekkel számszerűsíthetők? Néhány évvel ezelőtt egy szoftvert az készített el, aki valamilyen problémával találkozva úgy döntött, hogy azt számítógéppel akarja megoldani. Vagyis az, hogy egy szoftver jó-e, alkalmas-e a felhasználásra, aszerint dőlt el, hogy maga a készítő – és esetleg azok szűk köre, akik még használták a programot – elégedettek voltak-e vele. Az utóbbi időben azonban más a helyzet. A szoftverek mind nagyobb számú alkalmazó részére készülnek, a fejlesztőnek többnyire nincs is kapcsolata a felhasználóval. Azt, hogy egy szoftver mennyire jó minőségű, ma már elsősorban az határozza meg, hogy milyen széles körű a felhasználhatósága, mennyire hajlékonyan képes idomulni a különböző alkalmazói feltételekhez. Szakkifejezéssel élve: a mikrohatékonysággal szemben a makrohatékonyság került előtérbe.

Egy komolyabb szoftver kidolgozása nyilván nem kis időbe telik. Eredetileg nyilván valamilyen speciális feladat megoldása a célja. Viszont mire elkészül, könnyen lehetséges, hogy más igények is fellépnek az adott probléma kapcsán. Szükséges hát, hogy a szoftver alkalmas legyen a továbbfejlesztésre, ne kelljen elölről kezdeni a kidolgozást. Ezek alapján a szoftver minőségét meghatározó fő tényezők – ahogy Szentes kifejti: a jelenlegi állapotban való felhasználhatóság, a funkciók módosíthatósága, valamint a más hardver–szoftver környezetbe való átvitel lehetősége. A kötet első fejezete e tényezőket finomítja, bontja elemeire, valamint az ezek kapcsolatát, hierarchiáját leíró alapvető modelleket mutatja be.

A második fejezet a tényezők mérésének eddigi – rövid – történetéről ad áttekintést, és a gráfelméletből kölcsönzött fogalmak segítségével mérőszámokat rendel hozzájuk.

Harmadik fejezet: a program analízisének, elemzésének bemutatása. Fontos kiemelni, hogy Szentes a szoftver szerves részének tekinti a program dokumentációját – joggal. Értelmezése szerint a szoftvertermék két részből áll: az egyik a forrásnyelvű program, mely a gép számára készült, és a gép érti meg; a másik a dokumentáció a felhasználó számára, amit az alkalmazónak, az embernek kell értenie. A két alkotóelem együttes jó minősége biztosíthatja csak a teljes szoftver kvalitását.

A program funkcióanalízisének feladata, hogy kiszűrje a többszörösen szereplő, redundáns, illetve a végrehajtásra nem kerülő funkciókat, valamint, hogy meghatározza az egyes részfunkciók bonyolultságának mértékét.

Az eddigi három fejezet volt hivatva bemutatni a szoftverminőség-meghatározás alapfogalmait, azok értelmezését. A negyedik fejezet törést jelent ehhez képest. Szándéka szerint a minőségellenőrzés tevékenységeivel, illetve az ezzel kapcsolatos további tevékenységekkel foglalkozva – vagyis azzal, hogy hogyan is kell az eddigiekben leírtakat a gyakorlatban végrehajtani. Csakhogy az oldalakon hemzsegnek a törvényekre, rendeletekre, szabványokra való hivatkozások, melyek feladata e tevékenységek szabályozása. Az elvégzendő feladatok leírása agyonformalizálva, pontokba szedve olvasható. Remélhetőleg a szerző nem ezzel kívánja igazolni a szoftverminősítő eljárás szükségességét, létjogosultságát, illetve diszciplína-voltát – azt az olvasó az előzőek alapján úgyis vagy elhiszi, vagy sem.

Az ötödik fejezet ismét hasznos: az SZKI által kifejlesztett SOMIKA (szoftverminőség-ellenőrző és kapacitásfigyelő) rendszer használatát ismerteti, mely példát ad az első három fejezetben ismertetett módszerek számítógéppel támogatott megvalósítására. A rendszer megjelenésekor (1982) sok vonatkozásban a világon egyedülálló volt.

A könyv azoknak ajánlható, akik rendszeres felhasználói a legkülönfélébb alkalmazói szoftvereknek, vagy maguk is azok kidolgozásával foglalkoznak. A negyedik fejezet kivételével rendszerező elemzéssel – de nem szőrszálhasogatóan, nem elvont szakzsargont használva – mutatja be a szoftverek minőségét meghatározó tényezők számszerűsítésének lehetőségeit.

A „szoftver" fogalma alig tizenöt éves – az értékelésével foglalkozó tudományág nyilván még fiatalabb. A szoftver mérése kialakulóban, fejlődésben van. Nem tudni, mikor kristályosodik ki (kikristályosodik-e) annyira, hogy általánosan elfogadott és alkalmazott eljárások együttese legyen. Az első lépések mindenesetre megtörténtek a cél elérése érdekében.

Tallér József

MI HOGY CSINÁLJUK



–a számítógépes tábort?

A Borsod Megyei Tanács művelődési osztálya szervezésében 1986. június 30-tól, egy hétig, a miskolci 104. sz. Szakmunkásképző Intézet – Miskolc-Pereces – adott otthont a szakmunkástanulók számítógépes, bentlakásos táborának.

A megye minden szakmunkásképző intézete tájékoztatást kapott a tábor beindulásáról. Összesen 20 szakmunkástanuló vett részt a tanfolyamon, hogy gyarapítsa számítástechnikai ismereteit.

A számítástechnikai tábor iránt főleg az első éves szakmunkástanulók mutattak nagy érdeklődést.

A tavalyi évben a megyei tanács által szervezett számítógépes táborban, amelyen a gimnazisták és szakmunkástanulók együtt vettek részt, már megmutatkozott, hogy a szakmunkásképzősök körében is egyre nagyobb szerepet játszik a számítógép. Ezt mutatja az a tény is, hogy az ez évi táboron részt vevő hallgatók közül megyénkből egy diák 13. helyezést ért el a szakmunkásképzős tanulók Országos Számítástechnikai Versenyén. Ennek hatására ebben az évben két külön tábort szerveztek, az egyiket a szakmunkástanulóknak Perecesen, a másikat a gimnazistáknak Kazincbarcikán.

A perecesi táborban a diákok C 16-os, C 64-es és HT1080 Z iskolaszámítógéppel ismerkedhettek meg.

Délelőttönként a tanulók különböző témákból előadásokat hallgattak, melyen megszerezték elméleti alapismereteiket. A délutáni foglalkozásokon, számítógép mellé ülve, gyakorlatban hasznosíthatták a már megtanultakat.

Az elméleti ismeretek megszerzésében nagy segítséget nyújtott a megyében kiadott, Duzsa Árpád által írt „A HT 1080 Z iskolaszámítógép bemutatása programokkal" című könyve. A könyvben található programok nagy része C 16-os gépre is át van írva, így ebben a témakörben is jól használható volt. Ez lehetőséget adott arra, hogy a kétféle géptípuson futó azonos programok közötti eltérés és azonosság jól felismerhető legyen.

További segítséget jelentettek a C 16-os gép grafikus lehetőségeinek megismerésében egy megyei lap – az Észak-Magyarország – cikksorozatában megjelenő leírások és példaprogramok.

A gyakorlatok során kitűnően lehetett használni egy most kifejlesztett illlsztő egységet, melynek segítségével 6 db C 16-os gép tudott egy foppyval való üzemeltetésével csoportos foglalkozásokat is könnyen meg lehet oldani, így minden igényt kielégítően lehet az oktatásban használni.

A diákok előadást hallottak többek között az UNICOMP–3, univerzális interface készülékről, amelyet a miskolci Zalka Máté Gépipari Szakközépiskola fejlesztett ki és gyárt. Ez a készülék alkalmas több csatornán digitális és analóg jelek kiadására és fogadására, valamint feszültség, áram és ellenállás mérésére. Az UNICOMP–3 működésének bemutatására az előadó egy hőmérsékletmérést mutatott be, valamint ismertette a fordulatszámmérés, nyomásváltozásmérés, motorvezérlés és az egyes szakmáknak megfelelő vezérlési feladatok megoldásainak lehetőségeit.

További előadások keretében a BASIC programkészítés technikájával ismerkedtek a diákok.

A résztvevők között voltak olyanok, akik lényegesen több ismerettel rendelkeztek már, így a résztvevőket két részre osztottuk. Az egyik csoport a BASIC programozás alapismereteit bővítette, a másik csoport pedig a C 64-en történő programozás technikájával foglalkozott.

A C 16-os géppel dolgozó csoport a gyakorlat keretén belül feladatokat oldott meg, amelyet a gyakorlatot vezető tanárok értékeltek. A legötletesebb, leggyorsabban elkészült feladat szerzője jutalmat kapott.

A C 64-es csoport a megoldott BASIC feladatokon túl megismerkedhetett az EASY SCRIPT szövegszerkesztő programmal is. Ezt követően a megszerzett tudást egy konkrét feladat kapcsán a gyakorlatban hasznosították a diákok.

Egy másik alkalommal egy rajzolóprogrammal bővítették tudásanyagukat, amely nagy érdeklődést keltett.

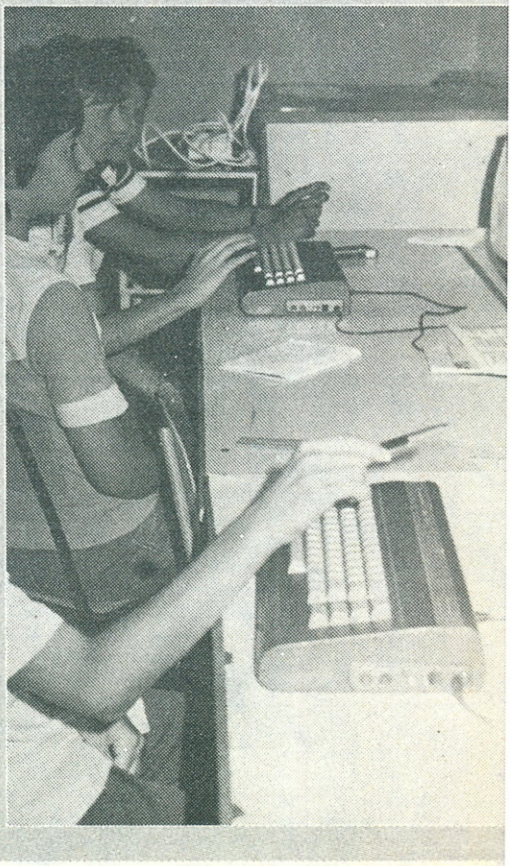
Az utolsó foglalkozás keretén belül a PRINT SHOP nyomtatóprogram kezelését sajátították el. A gyakorlatokat követően – vacsora után – a kollégiumban játékprogramokkal szórakozhattak a diákok, valamint programok cseréjére is sor került.

A tábor jól sikerült.

Az idő rövidsége miatt nem nyílt lehetőség egyes témákban mélyebb ismeretek szerzésére.

Nemcsak a tanulók, de a tanárok is értékes tapasztalatokat szereztek, amelyeket a jövőben, hasonló táborokban hasznosíthatnak majd.

Velezdi Tamás–Tárnoki Ilona



A szerkesztő azért van, hogy a lap olyan legyen, amilyenek az olvasói!

POSTA

Tisztelt Szerkesztőség!
Mindig érdeklődve és nagy figyelemmel olvastam a Kínpadon a... rovatot lapjukban. Sajnálattal vettem észre, hogy ez a rovat megszűnt létezni.
Kérésem a következő lenne:
Ha erre lehetőségük van, kérem, ezután is közöljenek le lapjukban hasonló rovatot, egy kis változtatással. Az eddigi közhasználatban lévő gépek (pl. Sinclair, Commodore) helyett a köztulajdonban nemigen szereplő nagyteljesítményű számítógépeket mutasson be a lap, ha ez nem oldható meg vallatási szinten, akkor ezekről a gépekről legalább ismeretterjesztő írások lássanak napvilágot. Ezek az írások is színesítenék az amúgy is színvonalas BIT-LET-et.
Kérem válaszoljanak, kérésem megoldható-e valamilyen formában.
Köszönet: ***Kapolka Gábor****, 7400 Kaposvár, Honvéd u. 45.*

A Vallató nem szűnt meg!
Rövidesen ismét jelentkezik a TVC vallatásával! Ötlete (vagy BIT-LET-e) nagy gépek bemutatásáról – érdekes. Megfontoljuk! – A szerk. –

Kérem, segítsenek nekem! Önök az 1985. 10. 31-i számban a 2 gépnyerőben feladatul egy szövegszerkesztő program elkészítését tűzték ki. Ebben a kiírásban szerepelt a következő mondat:
„Ismerünk néhány »Profibb« szövegszerkesztő programot HT-re...”
Én már régóta keresek HT-hez szövegszerkesztő programot, de sajnos sehol sem tudtam találni. Ez a mondat azonban új erőt öntött belém. Nagy hévvel kezdtem újra kutatni, ám az eredmény ismét csak semmi! Ezúton kérem önöket, küldjenek egy ilyen programot, vagy ha ez nem áll módjukban, közöljék velem. Régóta keresem a c-FORTH nevű programot is leírással. Segítségüket előre is köszönöm.
Kaszás Tibor*, 1173 Budapest, Újlak u. 102.*

Programot valóban ismerünk, de küldeni nem tudunk. Remélhetőleg az olvasók segíteni fognak.

Relatíve elég régen vagyok lapjuknak eleinte olvasója, majd előfizetője is. Sajnos most több Spectrum-os sorstársammal arra a megállapításra jutottunk, hogy lemondunk az előfizetésről, mert nem állunk olyan jól anyagilag, hogy „Commodore-centrikus” folyóiratokat is vásároljunk, illetve azokra előfizessünk akkor, amikor a Sinclair témák mind csökkenőbb arányokban fordulnak elő, sőt lassan mind gyakrabban hiányoznak is. Gondolom, nem vigasz az sem, hogy nem csak az önök lapjával szemben jutottunk erre az elhatározásra, hanem még 2 másik lappal is, mert teljesen felesleges olyasmikre pénzt kiadni, amit, ha egyikünk egy példányt megvesz belőle, akkor tapasztalhatjuk, hogy „Sinclair-szempont”-ból nem érdemes megvenni és elteni. Az eddigi tapasztalatok szerint így igen kevés példányt fogunk megvásárolni feleslegesen. Természetesen nem fegyverezni akarok ezzel, csupán a saját anyagi helyzetünkön kívánunk segíteni, hiszen önöknek egy pár spectrumos elvesztése nem jelenthet problémát. Csak azért talán érdemes lenne foglalkozni a gondolattal, hogy tekintettel az országban folyó „Commodore-kampányra” és a tervezett Commodore Újságra, na meg arra is, hogy nálunk reklámozni „kell” a sok megvásárolt és raktáron volt gépet – nem indokolt a jövőben saját sajtóval is rendelkező tábort, a többiek rovására támogatni.
Abban a titkolt reményben is ragadtam tollat, hogy talán majd egyszer újra érdemes lesz megvásárolni minden példányt és akkor újra előfizethetünk is rá (rájuk).
Reagálásukat szeretnénk oly módon tapasztalni, hogy újra a régi, nem részrehajló lapot vehetnének kezünkbe, addig is a magam és több Spectrum-os kollégám nevében búcsúzom lapjuktól: ***Képíró Róbert****, 1093 Budapest, Imre u. 5. 1/14. 378-075*

Kedves Sinclair-hívők!
Kedves Képíró Róbert!
Önkritikusan be kell ismernünk, hogy önnek igaza van!
Lapunkban valóban túlsúlyba kerültek a Commodore-anyagok, s a Sinclair-anyagok száma talán még a Primo-anyagokénál is kevesebb!
Mentségünkre alapelvünk szolgálhat, hogy tudniillik: a szerkesztő azért van... stb...
Az utóbbi hónapokban pedig nagyon ritkán jelentkeztek közlésre érdemes anyagokkal a Sinclair-hívők. A lehetőség pedig továbbra is lehetőség, ami közlésre érdemes, azt géptípustól függetlenül megjelentetjük.
Az igazsághoz az is hozzátartozik, hogy a C 16-os gépek tömeges megjelenése óta fontos feladatunknak érezzük, hogy lapunk anyagot adjon e gépekhez.
Végül ígérjük, hogy a jogos kritikát megszívlelve igyekszünk nem elfelejtkezni a nem kis Spectrumos táborról! Példa erre ez a lapszámunk is.
Képíró Róbert lapunkat még olvasó ismerőseit pedig kérjük, hogy juttassák el hozzá e lapszámunkat, hátha már tényleg nem olvassa!
Üdvözlettel: A szerkesztő

Tisztelt Szerkesztőség!
...A nemrég megjelent Szuper BIT-LET magazinból egy cikk különösen felkeltette a figyelmemet. A címe: Beszélő ZX. De sajnos a kis beszélőprogram ZX81-re van írva, úgy gondoltam, ha ezt a kis gépet meg lehet tanítani beszélni, az enyémet miért ne lehetne. Kérem önöket, küldjenek nekem egy, a fentiekhez hasonló, de Spectrumra írt beszélőprogramot. Fáradozásukat előre is köszönöm.
Minczér Csaba*, 3353 Aldebrő, Arany János út 68.*

Minczér Csaba levelének egy-egy részét idéztük, az elején írt dicsérő szavakat köszönjük.
A fent említetthez hasonló, de Spectrumra írt program létezik, neve SPEAKEASY.
Sajnos ez a program csak 48 K-s gépen fut, így csak azt tudtuk tenni, hogy a ZX81-re írt program szerzőjének címét levélben elküldtük. (A levélből az derül ki, hogy levélírónknak 16 K-s Spectrumja van.)
(Szerkesztőségünk egyébként sem vállalkozik programok küldésére, legfeljebb a „CSEREBERE” rovatban tudjuk a csereajánlatokat közzétenni.)

A BIT-LET-ben közölt Primo-térképet kipróbáltam, azonban a CTR + ↓ + valamelyik karakter lenyomása korántsem az ott közöltek szerint reagál. Primo A 64, P 84 001335/3 gyártmányú és gyártási számú gépemnél pl. CTR + ↓ +s lenyomásakor a μ (kis görög mü) jelenik meg, a RETURN billentyű lenyomásának hatására pedig OK felirat, és a helyőr a sor elejére kerül. De pl. a CTR, a ↓ és a q betű hatására ~ jelenik meg, és a RETURN lenyomására UL Error felirat a válasz. Előfordul továbbá zárójel, integráljel stb. csak az nem, ami az önök által közölt táblázatban szerepel. Hasonló a tapasztalat a SHIFT, a ↓ és valamelyik karakter lenyomásakor. A CTR vezérlőkódok CHR$ jelentése megegyezik a közöltekkel.
Érdeklődöm, minek tulajdonítható a tapasztalt eltérő viselkedés.
dr. Szentiday Klára*, főiskolai docens, 1013 Budapest, Attila út 25.*

Az ön Primo A-64-es gépe úgy „viselkedik”, ahogy az egy jól nevelt géptől elvárható.
A cikkben közöltek a programbeírásnál használhatók. Nézzünk egy példát:
Ha a CTR + ↓ +s gombokat „simán” lenyomja, akkor valóban a μ jelnek kell megjelennie. Ha azonban egy programsor bevitelekor a sorszám leírása után, a sor közben nyomja le a cikkben megadott billentyűkombinációkat, akkor – szintén az előbbi példánál maradva a μ jel jelenik meg, de a tárba a REM utasítás tárolódik el. Erről legegyszerűbben úgy lehet meggyőződni, hogy a sor lezárása (RETURN billentyű) után a LIST paranccsal ki kell íratni az éppen bevitt sort.
Ezen trükk tehát a hosszabb kulcsszavak egyszerű, gyors bevitelére ad lehetőséget.
Primo gépének használata során további sok sikert kívánunk.

Az 1986. májusi Commodore Újságban olvastam a GOTO X megvalósítását BASIC programból C–64-en. Egy hasonló ötletet valósítottam meg én is PRIMORA, amelyet szívesen közreadok.
A program során meg kell adni az X változó értékét és GOTO 3 utasítást. A 3-as sor átírja a 2-esı, és utána erre ugratják a programot.
(Pl.: 15 X = 40:GOTO 3)
Az 1-es és 2-es sorokban a szóköz fontos!
Molnár Tibor, Nemesapáti, Fő u. 17. 8923

```
1 GOTO 5
2 GOTO :::::::.
3 N1=INT(X/10000):N2=INT(X/1000)-N1*10:N3=INT(X/100)-N1*100-N2*10
:N4=INT(X/10)-N1*1000-N2*100-N3*10:N5=X-N1*10000-N2*1000-N3*100-N
4*10:POKE 17400,N1+48,N2+48,N3+48,N4+48,N5+48,0:GOTO2
5 REM: ITT KEZDÖDHET A PROGRAM
```

SZOFTVER ÖTLETEK



A Bit-let júniusi számában bukkantam rá Szabó Gál András C64-re írott fényújságjára. Mivel pont ez kellett, igen megörültem neki. Rögtön kipróbáltam, s remekül működött.
Sajnos rögtön kiütközött a program egy korlátja, a max. 255 karakteres szöveghosszúság, s az, hogy a szöveget sehogyan sem tudtam a képernyő tetszőleges helyére átvinni. Ezért Szabó Gál András ötletéből kiindulva írtam egy másik fényújságprogramot, ami egészen jópofa dolgokat tud produkálni. Akár egy kisregény is futtatható a fényújságon, a képernyő tetszőleges helyén, és az „újság" hossza is tetszőleges. Érdemes kipróbálni!
A program lényegét a 3030-as sorban lehet megtalálni, és máshol is felhasználni, mert ezzel a sorral lehet tetszőleges helyre pozicionálni egy szöveget.
Szőnyi Péter, Törökszentmiklós, Mártírok útja 161.

```
1 REM *** COMMODORE-64 FENYUJSAG ***
10 PRINTCHR$(147):POKE53281,0:POKE53280,0
40 X=9 :Y=5 :N=20 :D=13
60 PRINT"■■■■            ┌──────────────────────┐"
70 PRINT"                │                      │"
80 PRINT"                └──────────────────────┘"
90 GOSUB3000
100 END
110 :
120 :
3000 FORS=1TOD:READX$
3010 A$=RIGHT$(A$,N)+X$
3020 FORI=1TOLEN(A$)-N
3030 POKE214,Y:POKE211,X:SYS58640
3040 PRINT"■"MID$(A$,I,N);
3050 FORK=1TO80:NEXTK,I,S
3060 RETURN
3070 :
3090 :
3100 DATA"                      ****** "
3110 DATA"E RUTIN SEGITSEGEVEL ADOTT SZOVEGET LEHET A KEPERNYORE FENYUJSAG "
3120 DATA" SZERUEN, TETSZOLEGES HELYRE ES SZELESSEGBEN KIIRATNI! "
3130 DATA"MINT A KEPERNYON IS LATHATO, A SZOVEG KERETBEN IS FUTTATHATO. "
3135 DATA"A SZOVEG, A KERET ES A HATTER SZINE SZABADON MEGVALASZTHATO. "
3140 DATA" A SZOVEGET ''DATA'' SOROKBAN KELL MEGADNI. A RUTIN AZ EGYES DATA "
3150 DATA" SOROKAT FOLYAMATOS SZOVEGGE FUZI OSSZE. A KEPERNYO NEM VILLOG, "
3160 DATA" ES A SZOVEG JOL OLVASHATO. ---- A RUTIN A KOVETKEZO ADATOKAT KERI:"
3230 DATA" ''X'' A FUTO SZOVEG KEZDETENEK TAVOLSAGA A KEPERNYO BALOLDALATOL --"
3240 DATA" ''Y'' A SZOVEG TAVOLSAGA A KEPERNYO TETEJETOL. -- ''N'' A FUTO SZOV"
3250 DATA"EG HOSSZUSAGA -- ''D'' A DATA SOROK SZAMA.              A RUTIN BARME"
3260 DATA"LY PROGRAMBA BEEPITHETO!   ---     HASZNALD EGESZSEGGEL!   *****"
3270 DATA"                    "
```


Gépnyerő

SZUPER BIT-LET

A SZUPER GÉPNYERŐ (HOMELAB 3) ÉRTÉKELÉSE

88 megoldás érkezett, s ezekből 22 volt teljesen hibás. A maradékkal igen nagy bajban voltunk, a pontos bizonyítástól a nyilvánvalóan helytelen és hiányos bizonyításig mindenféle akadt, olyannyira, hogy sorba rakva a bizonyításokat, két egymás utáni között általában csak néhány szó, vagy néhány félmondat eltérés volt. A rangsorolást elsősorban az első feladat bizonyításának egyetlen kényes pontja szerint végeztük, ami annak indoklása volt, hogy két bőröndben együttesen mindig több, mint 6 kg súly van. Mivel ez volt a bizonyítás kulcslépése, ezért akik ezt a tényt indoklás nélkül leírták, bizony hátra kerültek a rangsorban. Szintén hátrább kerültek az olyan indoklások is, hogy: „nyilván nem lehet két bőröndben 6 kg súly vagy annál kevesebb, hiszen akkor azokat a gyerekek egy bőröndbe pakolták volna". Ez így nem teljesen igaz, nem pontos. Még rosszabbak voltak az olyan bizonyítások, melyek egy kész pakolást (pl. a mamáét) „rendezgettek" át, utólag ezt pakoltatták át a gyerekekkel. Ez teljesen téves szemléletű megfogása a dolognak, több pályázónk ugyanilyen elven „bebizonyította", hogy $B(L) <= 1.5 * M(L)$ mindig teljesül, s így a 4. feladat megoldhatatlan. Pedig, mint utaltunk rá, lényegében a legrosszabb esetekben Benedek 1,7-szer annyi bőröndöt is felhasználhat, mint a mama. Tehát szó sincs arról, hogy a gyerekek egy kész (főleg optimális) pakolást rendezgetnének át, ők egy listát pakolnak be egy jól meghatározott algoritmus szerint (ezért az M(L)-re vonatkozó indukcióval történő bizonyítások is hibásak voltak). Legszebbnek az olyan bizonyításokat ítéltük, melyek egyben az első házi feladatot (lásd alább!) is bizonyítják. Mindezek alapján úgy döntöttünk (hosszas meditálás után), hogy nem húzunk jó-rossz határvonalat, nem azt mondjuk, hogy ez a 29 bizonyítás jó, a többi 37 rossz, hanem a 20 legjobb megoldást beküldő pályázónk között sorsoljuk ki a gépet. Lehet, hogy ez nem tökéletes megoldás, de jobbat nehéz lenne kitalálni, s pályázóinkkal szemben is ezt érezzük a legtisztességesebbnek. (Már máskor is előfordult – s még valószínűleg elő is fog fordulni –, hogy a megoldásokat ismerve, azok színvonala alapján döntjük el, hogy kik között és hogyan sorsoljuk ki a gépet.

Tehát a 20 legjobb megoldást beküldő pályázó:

Adamát Lajos – Kecskemét, Baló Árpád – Győr, Bátki Zsolt – Budapest, Böjthe Zoltán – Budapest, Csikós Zoltán – Kiskunfélegyháza, Fogler Tibor – Budapest, Gyarmati György – Sümeg, Kisné Kunkel Petra – Szeged, Koncz Károly – Budapest, Likár László – Baja, Mészáros Gyula – Budapest, Nemes Imre – Kötegyán, Nyéki Péter – Kemenesszentpéter, Pallagi László – Százhalombatta, Pupp Gábor – Pécs, Pöltl János Tamás – Tatabánya, Szabó Sándor – Budapest, Szász Károly – Sükösd, Tegzes Istvánné – Budapest, Varga József – Debrecen. Magára a sorsolásra szeptember 12-én pénteken 10 órakor kerül majd sor a BIT-LET szerkesztőségében.

„Házi feladat"

Ez amolyan továbbgondolásra érdemes adalék a feladattal foglalkozóknak. Beküldeni nem kell (csak megoldani).

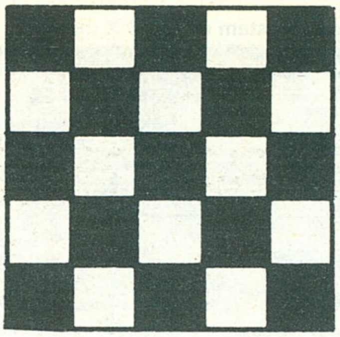
1. Nelli, Benedekék unokatestvére egy L listát N(L) bőröndbe pakol a következő módon: megfogja a soron következő holmit, s megnézi, belefér-e az utolsó bőröndbe (amelybe utoljára pakolt – az elején persze a legelsőbe), s ha igen, odarakja, ha nem, akkor a következőbe. Tehát Nelli még Benedeknél is gyorsabb, hiszen mindig csak 1 bőröndöt kell megnéznie. Bizonyítsuk be, hogy minden L listára $N(L) < 2 * M(L)$!

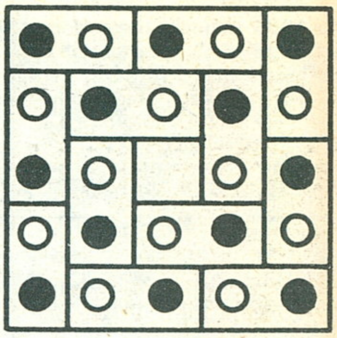
2. Bizonyítsuk be, hogy végtelen sok (tehát tetszőlegesen hosszú) L lista létezik, melyre $N(L) = 2 * M(L) - 3$!

GÉPNYERŐ

A Gépnyerő 1. feladatának megoldása

Első rész. Képzeljük el, hogy nem sima négyzetrácsos papíron játszunk, hanem egy „sakktáblaszerű" játékmezőn, azaz a négyzetek ki vannak színezve, felváltva feketére és fehérre (l. az ábrát!). Kezdetben Ferencke fekete, Félix fehér mezőkön áll, s a lyuk fekete mezőn van. Mivel minden lépés szomszédos mezőre történik, ezért a lyuk minden lépés után más színű mezőre kerül. Ebből következik, hogy Félix minden lépése előtt fekete mezőn lesz a lyuk, Ferencke lépései előtt pedig minden fehéren. Más szavakkal Félix mindig csak fehér mezőről fekete mezőre tolhatja figuráját, Ferencke pedig mindig csak fekete mezőről fehérre. Ebből az is következik, hogy a játék során minden figura legfeljebb egyszer mozoghat, hiszen pl. Félix valamely figurája, ha egyszer fehér mezőről feketére lép, akkor többet nem léphet, hiszen csak Félix tolhatná odébb, de mint láttuk, ő mindig csak fehér mezőről tolhat feketére bábut. Így mindjárt kiderült, hogy a játék véges, mivel mindkét játékosnak 12 bábuja van, s minden bábu egyszer mozoghat, így legkésőbb 12 lépéspár után a játék biztosan véget ér.





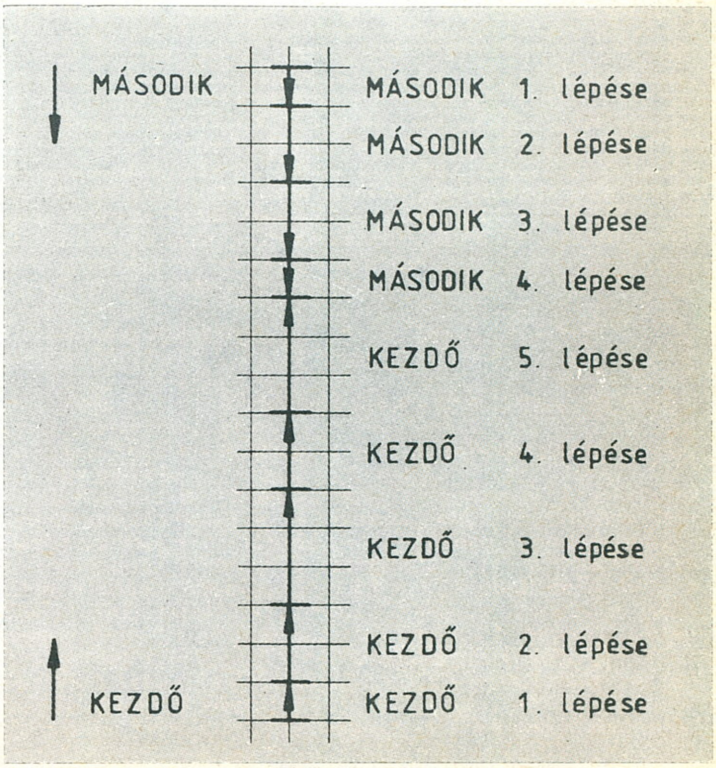
Második rész. Ezek után Ferencke nyeréséhez elegendő, ha biztosítjuk azt, hogy Félix minden lépésére tudjon válaszolni, hiszen így az előzőek miatt előbb-utóbb Félix lesz az, aki nem tud lépni. Egy ilyen stratégiát könnyen megadhatunk a 2. ábra segítségével, azaz, ha képzeletben 2×1-es dominókkal fedjük le az ábra szerint a játékmezőt. Látható, hogy induláskor minden dominóban egy fehér és egy fekete bábu van. Ha Félix valamelyik fehér bábut eltolja, akkor abban a dominóban egy fekete bábu és egy üres hely lesz. (Megjegyzés: Félix első lépésben középre, „dominókon kívüli helyre" tol egy bábut, ezt azonban az első részben megállapítottak miatt többet nem mozgathatja!) Legyen Ferencke stratégiája az, hogy mindig abban a dominóban lép, amelyikben a lyuk van lépése előtt, tehát abban a dominóban elhelyezkedő bábuját tolja az üres helyre. Az első ilyen lépése után fehér ismét rákényszerül, hogy valamelyik, még nem bolygatott dominóból kilépjen, s Ferencke újra tudja alkalmazni a stratégiát. Mivel Félix csak mindig olyan bábuval léphet, amit még nem mozgatott, s az ilyen bábuk dominójában levő fekete bábut Ferencke addig nem mozgatja, míg a fehér bábu mellőle ki nem lép, ezért látható, hogy e stratégia alapján Ferencke valóban Félix minden lépésére tud válaszolni, s így nyer.

harmad-gép nyerő



2. feladat

Megint egy jó játékról lesz szó. A játékot pl. négyzetrácsos papír egy vonalának előre meghatározott szakaszán játszhatja két játékos. A játékosok a „pálya" két különböző végéről indulnak, s első lépésként felváltva lépnek, a következő szabály szerint:
Mindig szabad 1-et „gyorsítani" vagy „lassítani", vagy ugyanolyan „sebességgel" haladni, mint az előző saját lépéskor, de megállni nem szabad. Pl. Kezdő első lépésével 1-et halad Második felé, majd Második is 1-et lép Első felé. 2. lépésre haladhat 1-et, vagy 2-t. Ha pl. 2-t haladt, akkor 3. lépésre haladhat 1-et, 2-t, vagy 3-at. Az nyer, aki „rálép" a másikra, vagy átlépi őt. (Lásd az ábrát, ahol Kezdő nyert.) A pálya hosszának nevezzük a kiindulási szakasz hosszát (ábránkon 17).
A feladat: bizonyítsák be, hogy 14 hosszúságú pályán Másodiknak, 15 hosszúságún pedig Kezdőnek van nyerő stratégiája!



Ápisz

Kérjük levágni és a borítékra felragasztani! Beküldési határidő: szeptember 22.

Szerkesztőségünk címe: ÖTLET Budapest 1986. Szerkesztő: Angyalosi László. Művészeti szerkesztő: Pribelszky Pál
Tördelő: Domokos Imre. Hírrovat: Csányi György. Vallató: Pogány György. Gépnyerő: Király Zoltán. Posta: Halász Péter